ファミリーコンピュータ™

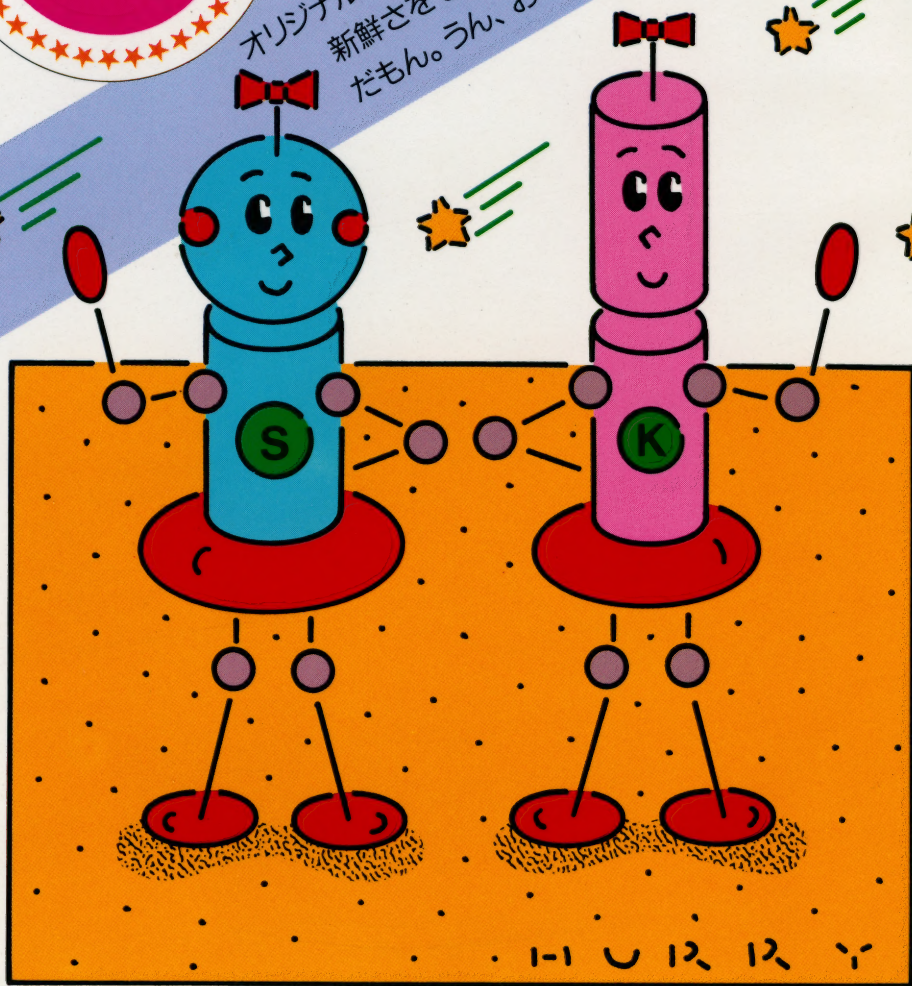
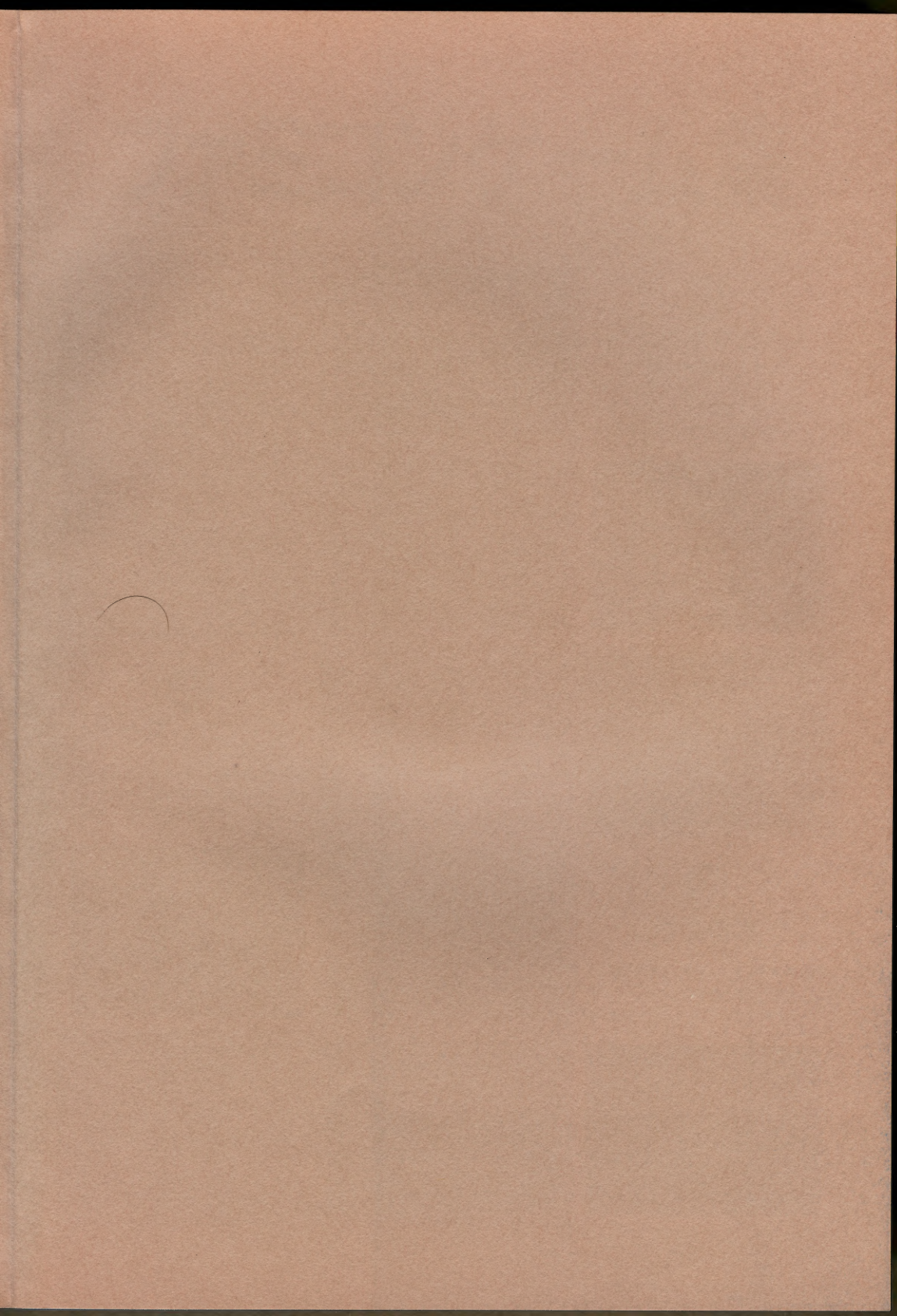
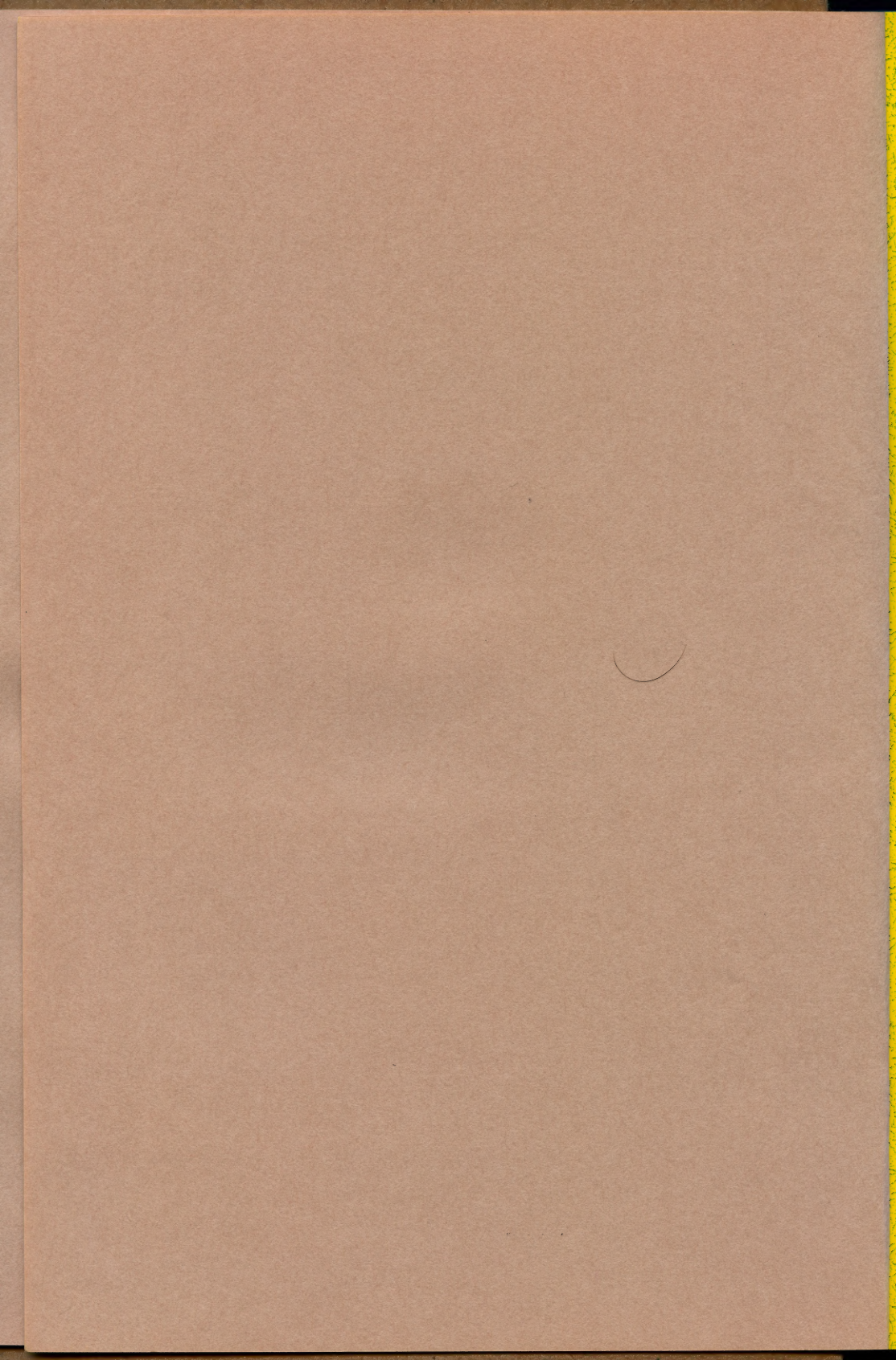
オリジナルゲーム集25

木村香奈枝・斉藤千秋著

オリジナルゲーム25+おまけ26=51本。
新鮮さをそのままパックしたんだもん。うん、おいしいネ♥

ファミリーコンピュータ オリジナルゲーム集25

木村香奈枝 斉藤千秋著

ナツメ社

ファミコンゲーム集

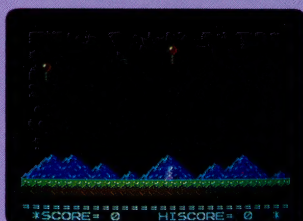
1 カニカニ星人襲来

『カニカニ星人』が，ゾロゾロカベを登ってくるんですー。つぎつぎと蹴っ飛ばして，落っことしちゃってネ。

2 不思議の国のバルーン

不思議の国にすむ『風せんクン』が逃げ出しちゃったの。はやくつかまえないと，竜になって暴れるんです。

3 ドキドキすごろくルーレット

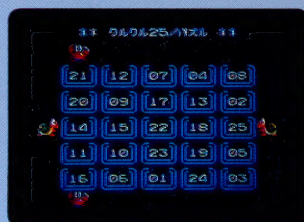
上手にルーレットを回して，『ルーレットの部屋』から出てくださいねー。

4 メイズ・パニック

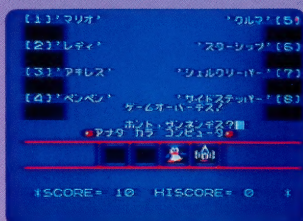
コンピュータルームに，爆発物がしかけられました。調査船に乗り込んで，回収しなくてはならないのです！

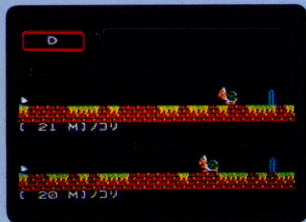
6 くるくる25パネル

バラバラになった1から25までの『数字パネル』を，元通りに戻してネ。カメさん，カニさんだって，大活躍しますぅ。

5 あなたの能力どれくらい？

コンピュータの出題するマークを，いったいいくつまで覚えられるかなー？

7 月面着陸

月面着陸船が，故障！ とにかく緊急着陸しなくちゃ…でもネ，着陸できるのは，谷底ふかーくにしかないんだもん。

8 よい子のかけっこ競走

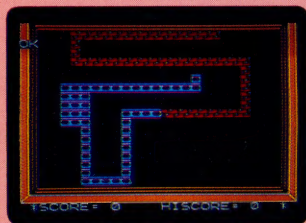
カメとウサギ…，じゃなくって。カメさんどうしの『かけっこ競走』です。キーボードを叩いて，走りますー。

9 散歩でよいしょ！

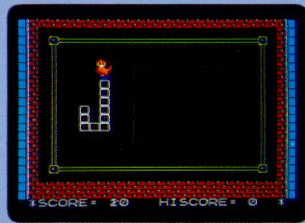
公園の中を，散歩します。しながら数字を拾ってネ。マァ，いわゆる『オリオンテーリング』なんですねー。ウン。

10 じゃんけん、ぽん

じゃんけん，ポン。あいこで，しょ！ コンピュータと『じゃんけん。するの。でも，あと出しはしないで…。』

11 あるけあるけゲーム

ふたりで，テレビ画面の中を歩き回るの。それで，先にカベとか足跡に，ぶっかっちゃった方が，負けなんですう。

12 ハッスルしましょう

『ヘビさん』がエイリアンを食べると，身長がのびちゃうー。だんだんのびてきちゃって，画面いっぱいになるんです。

13 ブロックくずし・スペシャル

両手を使って，ブロックくずし，します。右手と左手って，協力してくれないの。

14 マンホール大作戦

マンホールのフタが，無くなってます。このままだと，歩いてる人が落ちちゃう。フタをして，助けてあげてくださいネ。

15 ファイヤー救助隊

こんどは，ビルに火災が起きました。つぎつぎと，人が飛び降りてきますー。トランポリンで，救急車まで運んであげてネ。

17 バルーン・ボンバー

『熱気球』に乗って，世界1周。とまでいかないけど，小旅行です。途中には，山あり谷ありで，もう，タイヘン！

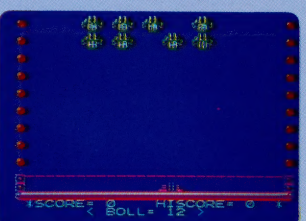
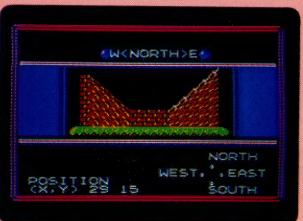
16 スロットマシーン

ガシャン，くるくる…，ジャラジャラ。スロットマシンです。この本格スロットマシンで，不敵なギャンブラーに！

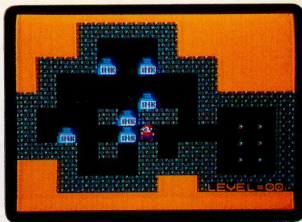
19 インベーダー

インベーダが，宇宙からゾロゾロと侵略してきました。地上で使えるのは，ミサイル砲だけ。しかも，12発しかなんて。

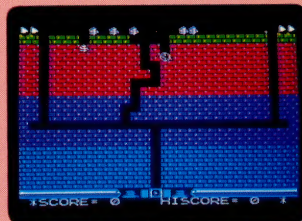
18 立体！三次元迷路

『立体迷路』の中に，入ったんです。出口を探してネ。これは，現実感がヒシヒシ伝ってくる，おすすめゲームですう。

20 倉庫番だって、タイヘンです！

食庫で，整理整とんのアルバイト。ヨッコラショ。て，かたずけてネ。

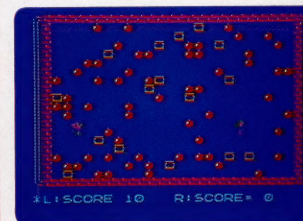
21 ディープ・スキャナー

海中を進む，潜水艦『Uボート』を爆雷を落としてやっつけます。深くにいるUボートの方が，高得点です。

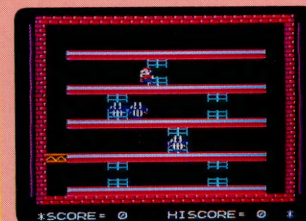
22 2001年地底の旅

2001年。地球のエネルギーのほとんどを地底から送ってました。これを狙って『エイリアン』が地底に潜ってきたんです。

23 お花畑で、いっしょにネ！

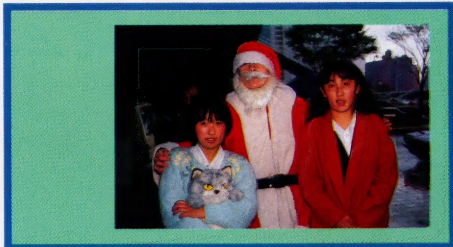
お花畑で，リンゴ狩りをします。でもね ふたりで，ケンカしないように…。

24 恐怖のエレベーター・エイリアン

5階建てのビルに侵入したエイリアンをエレベータを上手に使って退治して。

25 ヒョロロンとガラゴロン

『ガラゴロン』に食べられないように，逃げながらブロックで囲んでねー。

ふんわり、背のびしたいな

▽ワァ、ちーちゃんのぬいぐるみ，写真になっても，やっぱりカワイイ。▼そうでしょー。だって，わたしの，『お気にいり。』なんだもん。このジュニアの目って，大きさが右と左じゃあちがうでしょ？　そこが，気に入ったの。▽あ。わたしも，そう思う…。▼それと，えと。しっぽがネ，まるまる太ってるでしょー。▽うん。たぬきサン，みたいに…。▼そうなの。それでね，そこをナゼナゼってすると，気持よくって。▽ぷっ。あー，でも分かる。

▼これをネ，見てて思うんだけどぉ。▽ん。▼なんだか，わたしって，みーんな笑ってるのばっかり，でしょ？▽ほんと？　あ，ほんと，ちーちゃんのって，みんな笑ってる…。▼でしょー。でもぉ，ちょっぴりヘンじゃない？▽え？　どーして？　とってもカワイクって，ワァいいな，と思うのに…。▼だってぇ。これじゃあ，悩みなんてぜーんぜんない女のコ，みたいなんだもん。わたしだって，悩みくらい，ありますよぉ。えと，それにネ。笑ってるから，目がハニワみたいなんだもん。うー，クスン…。▽あっは。だいじょぶ，大丈夫。でも，いつものちーちゃんって，もっと目がくりくりっとしてる，のにね。ちょっぴり残念，かも。

▼わあ，サンタのおじさんも，ちゃーんと写ってる。▽あッ，ほんと。でも，ホントは，宣伝のおにいさん…。▼そうなの。でも，なんだかヘンな感じネ。▽そーね。▼あ。そういうば，突然，雪がたくさんあったりいして。もう，わたし，キャーキャー喜んじゃった。だって，突然，パッて雪があるんだもん。▽まだ，雪の季節じゃないのに，不思議。ちょっぴり，雪合戦したり…。▼いつの間にか，わたしの手，まっかになっちゃってたの。

▽えと。あとは…。▼あ，そういえば。かなえ，アイスクリームたべなかったでしょ。もう！，さむがり屋さんなんだからー。▽え，だって。すっごーく寒かったから，こごえそうに寒いのに，アイスたべよう！，なんて…，もう。▼でも，ああいう寒い日にネ，さむいよぉー，なんていいながらたべるのが，一番おいしいの。▽もう。口のなかが，しもやけになってもしらないから。でも…，ちょっぴり寒くなかった？　▼うん，ほんとはネ，さむかったの。

ファミリーコンピュータ
オリジナルゲーム集

発行————1986年1月15日
著者————木村香奈枝・斉藤千秋
発行者———田村正隆
発行所———ナツメ社
郵便番号101
東京都千代田区神田神保町1-52
電話〈03〉291-1257
〈落丁・乱丁本はお取り替え致します〉
定価————980円
カバー・絵——出原速夫
coverdesign—tomoko niwa

ファミリーコンピュータ™

オリジナルゲーム集

木村香奈枝・斉藤千秋著

ゲームは，オバケだったんです，実は…。

まえがき

ゲームは，『**オバケ**』**です！**

オバケって，今はもう，現実にはいないって分っちゃってるけど。でも，もしも，本当にいたら…。たぶんゲームは，このオバケみたいじゃないかナ？

どうして，オバケって怖いんでしょ。

いつ，出てくるか分らないから，**ハラハラ！**

どんなに恐しいカオをしているか分らないから，**ドキドキ！**

これが，何時に出てくるのか！どんなカオしてるのか分っちゃってたら，ぜーんぜん恐くないですよね。「ワーイ，やっぱり出てきたよー」なんて，反対に喜んじゃったりして…。

ゲームも同じ，だと思うの。何が，どんな風に出てくるのか，想像もできないから面白いんです。

これが，わたしの「**怖い**オバケ=**面白い**ゲーム説」…。

で，この本のゲームは。もちろん，『怖いオバケ』たちばかりです。いつ，どんなカオをしてるのか**想像もできないよーな怖いオバケ**が，25ヒキも，そろってますー！

しかも，BASICプログラムで作ってあるから，同じカオに飽きたら，いくらでも自由に変えられます。**いつでも，新鮮！**なゲームを遊べるんです。（新鮮だなんて，オバケなのに…）

それに，いろんなタイプのオバケが，揃ってます。

三次元の迷路にもぐりこんだり，潜水艦を退治しにいったり。地底にまで，もぐりこむんですからもう，すごいオバケさん！。もしも，よければ，みなさんも，こんなオバケと，つきあってみてくださいネ。

よろしく。

フォーナイト企画部
斉藤　千秋（ちあき）
木村香奈枝（かなえ）

第1章
ゲーム入力の仕方は…。

第2章
オリジナルゲーム集です。

第3章
サンプルゲーム集です。

1

第1章 ゲームの入力の仕方は

ゲームを遊ぶ前に，こうしてネ

ファミコン・ユーザーの常識です

えと，ゲームを遊ぶ前に準備することを，説明しますねー。でも，もう，みなさんは分ってると思うけど…。

次の順番で，準備してネ。

その1：電源を入れます。
その2：BGグラフィックを入れます。
その3：BASICプログラムを入力。
その4：RUNします。

これで，遊びたいゲームは動き出すんです。ね?　とっても簡単でしょ?

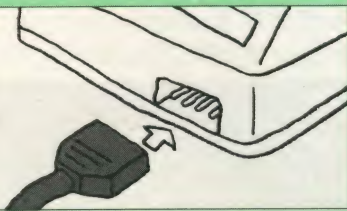
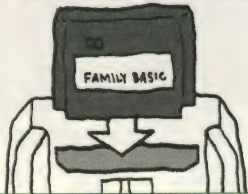
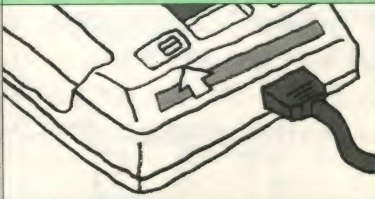
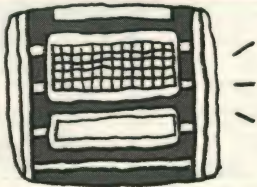
BASICプログラムのゲームって，自分で改良できるところがカートリッジよりいいもん。ゲームに飽きちゃたら，自分で面白くしちゃうのー。

こーいうのって，もう『ファミコン・ユーザーの常識』ですよね

電源をON!

『ファミリーコンピュータ』の電源をONにしちゃうには…。

①キーボードのコネクタを差し込みます。 ファミリーベーシック用のキーボードを本体にくっつけちゃう!　こーしないと動かないの。

②カセットを入れます。 ファミリーベーシックのカセットを入れます。この中に，BASICが入ってるんですよー。

③電源をONにします。 ファミコンの電源をONにします。

4[T]+[RESET]を押します。 [T]のキーと[RESET]のキーを同時にピョコって押してね。すると、『GAME BASICモード画面』になるんです。

これで，1を押せば『BASIC』に，2を押せば『BGグラフィック』になるの。間違っても3を押しちゃあ，だめなの。3を押しちゃうと，せっかく入れたプログラムとかBGグラフィックが消えちゃうんだもん。ひどいよおー!

電源をONにするには，こーするんです

1	キーボードを差し込みます
電源をＯＦＦにしたファミコン本体に，キーボードのコネクタを差し込みます．	

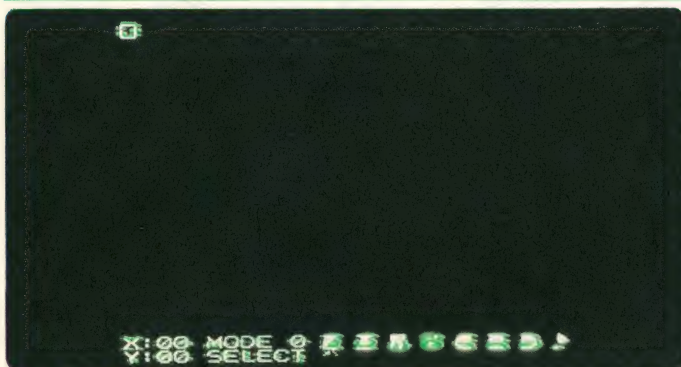
2	カセットを入れます
ベーシックのカセットをファミコンに入れます。キチンと入れてね。	FAMILY BASIC

3	電源をONにします
電源をＯＮにします。スタート画面がでてきますー。	

T ＋RESETを押します。「GAME BASICモード」になるんです。	

BGグラフィックの入れ方

GAME BASICモード画面で，てを押すと『BGグラフィックモード』になるんですー。BGグラフィックモードでは，**ゲームの背景を描きます。**

テレビ画面は，ちゃあんと次みたくなってます？

それでね，**BGグラフィック図**を見ながら，背景を描いてゆくんです。『キャラクターテーブルB』を見ながら，描きます。

たとえば，図のデータが〈K50〉てなってたら，Kグループの5のキャラクタを使って，色は0になるの。もちろん，キャラクタテーブルBを見なくちゃ，分りません。

それから，図のデータが〈A〉ってなってるときがあるの。こーゆーのは，CHARモードで直接キーボードから 'A' て入れればいいんです。

BGグラフィックの最初のモードはSELESTです。

▲ ▼ で選んで，
スペース で実行します。

ESC キーを押すと，『ファンクションメニュー』がでてくるんです。このファンクションメニューでは，画面をいろいろ編集したり，画面のデータをテープに保存したり読み出したりする，モードを選びます。

	ファンクションメニュ
SELECT	キャラクタを選びます。
COPY MOVE	画面内でキャラクタを編集します。
CLEAR	画面をキレイに消します。
FILE	テープにセーブしたりロードします。
CHAR	キーボードの文字を画面に表示します。

背景を描くには，このキャラクタテーブルBの図を見ます。

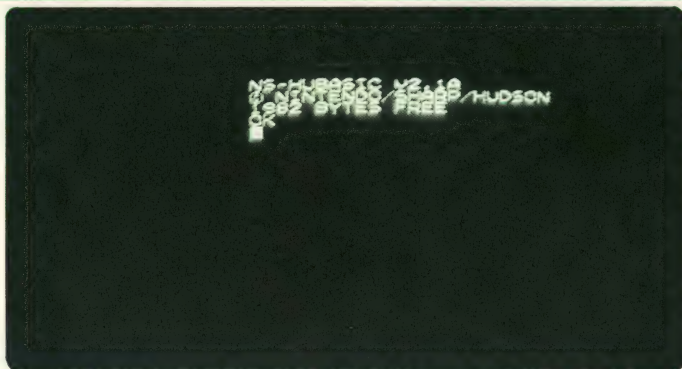
	Ø	1	2	3	4	5	6	7
A								
B								
C								
D								
E								
F								
G								
H								
I								
J								
K								
L								
M								

BASICプログラムの入力方法

BGグラフィックを入れ終わっちゃったから,今度は**BASICプログラムのばん**です。

BASICとプログラムを入力するには,『BASICモード』じゃなくちゃ,だめなの。だから,BASICモードにしておいてネ。

GAME BASIC画面モードの時に,〈1〉を入力すると,次のような画面になってBASICが動くんですー。

BASICが,動いたところ。

うわー,ちゃあんと,なっちゃった。これであとは,BACIC プログラムを入力するだけなんです…。

それじゃあ,入力の仕方を説明しましょう。オッホン!

BASICプログラムは,『行』がたくさん集まったものなんです。それで,入力するときは『行』ごとに [RETURN] キーを押して,入れてゆきます。

プログラム

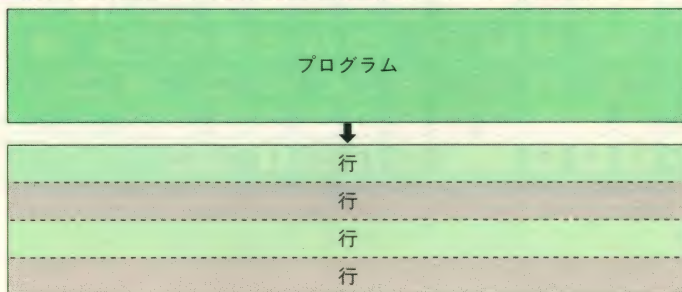
↓

行

行

行

行

プログラムは,行がたくさん集まったものです。
行は,行番号といくつかの命令でできてる,プログラムの基本単位です。

うーん。それからね，プログラムの中のREM文（’）のコメントは，入力しないでくださいネ。コメントまで入れると，**メモリが足りなくなって**，エラーになっちゃうんです。

あっ。でもREM文を取っちゃ，だめなの。**REM文のコメントだけを**取って入力してね。

```
100  '●●  GAME/01(チアキ+カナエ)  ●●
110  VIEW:SPRITE  ON
```

```
100  '←———コメントを取りません。
110  VIEW:SPRITE  ON
```

でもー，これだけじゃあ分らないかもしれないから，次のプログラムを，わたし，入力してみました。参考にしてネ。

```
100  '●●  SMPLE01(チアキ+カナエ)  ●●
110  FORX=1TO10
120  COLORX,10,2:LOCATEX,10:P
RINTCHR$(207);
130  IFSCR$(X+1,10)<>"+"  GOTO
150
140  NEXT
```

プログラムの中にREM文があったら，そのコメントは入力しないでくださいね。それから。プログラム・リストに〈カ″〉っていうふうになってたら〈ガ〉のことです。
わたしの使ってるプリンタが，濁点付きの字を出せないから，しかたないんです。

第2章

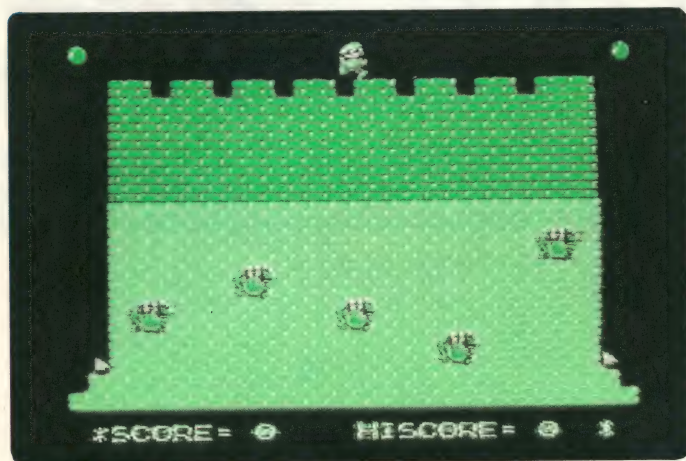
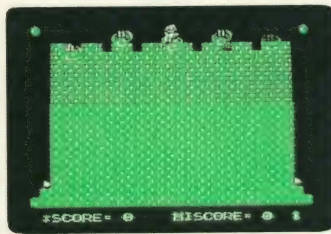
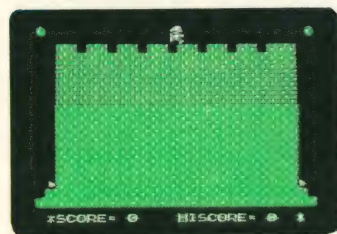
オリジナルゲーム集です

つぎつぎとカニ星人が登ってくる，イライラアセアセ，ゲームです。

1 カニカニ星人襲来!!

ゲームの遊び方

『カニカニ星人』が，壁をよじ登って攻めてきます。でもお，攻めてくるのにどーして壁を登らなきゃいけないんでしょ？

とにかく，よじ登る前に蹴っちゃってくださいネ。ぞくぞくとカニカニ星人が登ってくる，スリルとサスペンスのゲームですっ！

テレビ画面の下の方にいるのは、カニカニ星人です。よじ登ってくるのー。画面の上にいるのは，マリオです。

カニカニ星人を蹴っ飛ばしてくださいネ。レンガの上まで登ると，ゲームオーバ。

ゲームを遊ぶ前に，BGグラフィックを入れて。それから，BASICプログラムを入力してネ。

このゲームは，カニカニ星人が壁をよじのぼる前に蹴っ飛ばして落とすんです。蹴っ飛ばすには，カニカニ星人の上に『マリオ』を移動させれば，自動的になります。でも，星人が**色の変ったレンガ**の上にいないと，だめなのー。

カニカニ星人に，壁をのぼられちゃうとゲームオーバーです。

●200
変数の設定をしています。配列D(I)の値をぜんぶ176にします。

●210
カニのスプライトを設定してます。

●300~320
マリオの動きを決めています。コントローラの押した通りに，マリオが動くようになっています。変数Cはマリオが歩いてみえるようにスプライトを替えるための変数です。

●330~350
マリオの向きによって，スプライトをDEF SPRITE文で設定しています。変数VXの値が－1のとき左向き，1のとき右向きです。

●400~430
カニを動かします。配列O(I)の値が176のときは，カニは表示しません。176よりも小さいと，表示したり動いたりします。

●600~630
乱数の値が0のとき，配列D(I)の値を168にします。こうすると420行のIF文の条件に合うので，カニが動き出すようになります。

●700~760
カニを動かしています。

```
100  '■ カニカニセイシ゛ン   CHI+KANA ■
110  DIM D(4),M$(1)
120  VIEW:SPRITE ON
200  FOR I=0 TO 4:D(I)=176:NE
XT:S=0:X=13:VX=-1
210  M$(0)=CHR$(192)+CHR$(193
)+CHR$(194)+CHR$(195):M$(1)=
CHR$(196)+CHR$(197)+CHR$(198
)+CHR$(199)
250  LOCATE 0,22:PRINT" *SOCR
E= 0      HISCORE=";H;"    *"
300  I=STICK(0)
310  IF I=2 AND X> 0 THEN X=X
-1:C=C+1:VX=-1
320  IF I=1 AND X<26 THEN X=X
+1:C=C+1:VX= 1
330  D=(C MOD3)*4
340  IF VX=-1 THEN DEF SPRITE
0,(1,1,1,0,0)=CHR$(D)+CHR$(D
+1)+CHR$(D+2)+CHR$(D+3)
350  IF VX= 1 THEN DEF SPRITE
0,(1,1,1,1,0)=CHR$(D+1)+CHR$
(D)+CHR$(D+3)+CHR$(D+2)
360  SPRITE 0,X*8+16,28
400  FOR I=0 TO 4
410  IF D(I)=176 GOSUB 600
420  IF D(I)<176 GOSUB 700
430  NEXT
450  GOTO 300
460  '
600  '■サフ゛ルーチン1■
610  IF RND(100-S/5)>0 RETURN
620  D(I)=168  ←カニ星人の位置を決めます
630  RETURN
640  '
700  '■サフ゛ルーチン2■
710  D(I)=D(I)-4
720  DEF SPRITE I+1,(2,1,0,0,
0)=M$( (D(I)MOD8)/4 )
730  SPRITE I+1,I*40+40,D(I)
740  IF D(I)<80 GOSUB 800
750  IF D(I)<24 GOTO 900
760  RETURN
770  '
800  '■サフ゛ルーチン3■
810  IF X<>I*5+3 THEN RETURN
820  S=S+10:DEF SPRITE I+1,(1
,1,0,0,0)=CHR$(176)+"。[]"
830  LOCATE 10,22:PRINT S
840  SPRITE I+1,I*40+40,D(I)
850  PAUSE 20:SPTITE I+1
860  D(I)=176
870  RETURN
880  '
900  '■ケ゛ームオーハ゛ー■
910  LOCATE 9,10:PRINT"*ケ゛ームオー
ハ゛ー*":INPUT A$
920  GOTO 200
```

●フローチャート

RUN
100 ▶プログラムの注釈
110−210 ▶変数の設定と画面モードの設定をします
250 ▶画面作りをします
300−320 ▶コントローラ操作により，マリオの位置や向きの変数の値を求めてます。
330−350 ▶表示するアニメキャラクタの向きを決めてます
360 ▶マリオを表示
I=0 TO 4
カニを表示してない
サブルーチン1
カニを表示してない
サブルーチン2
NEXT
サブルーチン1
乱数値>0
620 ▶カニを表示へ一
RETURN
サブルーチン2
710 ▶カニ一歩上へ登る
720−730▶カニを表示します
カニが青い所にいる
サブルーチン3
カニが上まで登った
ゲームオーバー
RETURN
サブルーチン3
カニをマリオがけった
820−850▶スコアを表示
860▶配列D(I)に176を入れて，カニを表示しないようにします
RETURN
ゲームオーバー
910 ▶ゲームオーバー
GOTO 200

●変数とは？
BASICのプログラム中で使われる値を格納するためのエリアに，英数字から成る名前を対応させたものが変数です。
変数の値は，プログラムによって定義され，演算，参照などに使うことができます。ファミリーベーシックでは，文字変数と数値変数の2種類があります。

プログラムの説明

このプログラムでは，MOVE文の代わりにSPRITE文を使ってみました。

このプログラムは，マリオ君を動かすところがとても難しかったんです。もう，悩んじゃって…。

300行から360行までのプログラム，です。

ここのところでは，コントロールキーを押すとマリオ君が左右にトコトコ歩くように，してます。このトコトコ歩くように，が，とても大変でした。

ただマリオが左右に移動するだけなら，次のようにします。

```
10  SPRITE  ON:X=13
20  DEF  SPRITE  Ø,(1,1,Ø,Ø,Ø)=
CHR$(Ø)+CHR$(1)+CHR$(2)+CHR$
(3)
50  I=STICK(Ø)
60  IF  I=2  AND  X> 1  THEN  X=X-1
70  IF  I=1  AND  X<25  THEN  X=X+1
80  SPRITE  Ø,X*8+16,4Ø
90  GOTO  5Ø
```

●数値変数とは？
数値変数の変数名は，XやYなどの英数字記号で名前をつけます。変数名は255文字までの記号が扱えますが，コンピュータは最初の2文字だけで変数名を区別します。
また最初の1文字目の記号は，必ず英文字を使います。ただし，英文字であってもBASICで使う命令は使えません。

一応，簡単にプログラムの説明，しておきますね。

20行では，番号0のスプライトに**〈マリオ(WALK1)〉のアニメキャラクタを設定**してます。えと，この〈マリオWALK1)〉は，キャクタテーブルAの左上にあるアニメキャラクタのことです。

50行では，**変数I**にコントローラの押した値が入るように，してます。コントローラを左に押すと2，右に押すと1の値になります。

60行では，**コントローラが左に押されると**変数Xの値から1を引くようになってます。でも，変数Xの値が1より大きくなくちゃいけないんです。

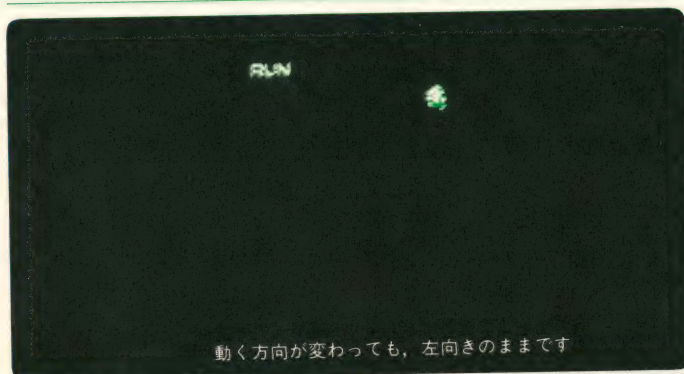
70行では，**コントローラが右に押されると**，変数Xの値に1をたすようになってます。

それで**80行**で，変数Xの位置に**マリオを表示**してます。

では，このプログラムを実行してみます…。

```
RUN (RETURN)
```

動く方向が変わっても，左向きのままです

●文字変数とは？
文字変数の変数名は，XやYなどの英数字記号の後に〈$〉ダラーマークを付けて数値変数と区別します。変数名は数値変数と同様に255文字までの記号が扱えますが，コンピュータは最初の2文字だけで，変数名を区別します。また，最初の1文字目は，必ず英文字を使います。

どうです？。ちゃんとコントローラ操作の通りに，マリオが左右に動きます。でも…，マリオは左向きのままですし，それにトコトコなんて動かないんです。

これじゃあ，なんだかつまらない，です。

そこで今度は，ちゃあんと右側へ動いてる時は右向き，左側へ動いてる時は左向きになるようにしてみました。

```
10  SPRITE ON:X=13
50  I=STICK(0)
60  IF  I=2  AND  X> 1  THEN  X=X-1
:VX=-1
70  IF  I=1  AND  X<25  THEN  X=X+1
:VX= 1
72  IF  VX=-1  THEN  DEF  SPRITE
0,(1,1,0,0,0)=CHR$(0)+CHR$(1
)+CHR$(2)+CHR$(3)
74  IF  VX= 1  THEN  DEF  SPRITE
0,(1,1,0,1,0)=CHR$(1)+CHR$(0
)+CHR$(3)+CHR$(2)
80  SPRITE  0,X*8+16,40
90  GOTO  50
■
```

このプログラムでは，マリオの向き用に変数VXを用意しました。変数VXが－1ならマリオは左向き，1なら右向きです。

60行では，コントローラが左に押されると，マリオを左に動かして，向きを左向きにします。

70行では，コントローラが右に押されると，マリオを右に動かして，向きを右向きにします。

BGグラフィック画面01

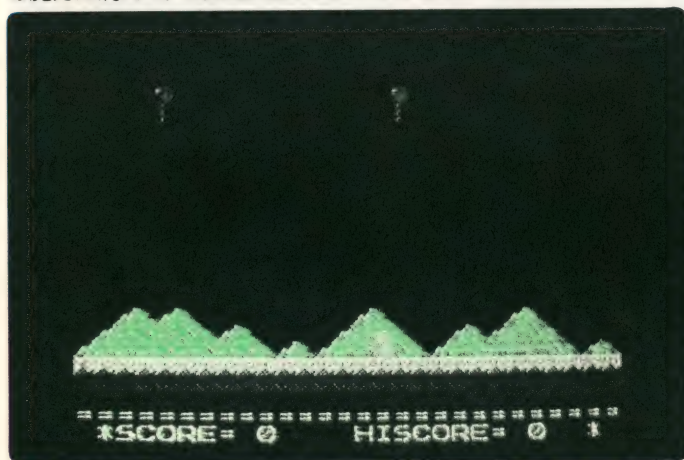
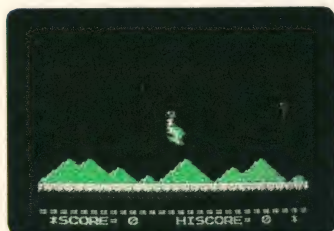
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
0	H72																											H72
1																												
2			F30	F30		F30	F30		F30	F30		F30	F30		F30	F30		F30	F30		F30	F30		F30	F30	F30		
3			F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30		
4			F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30		
5			F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30		
6			F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30		
7			F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30		
8			F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30		
9			F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32		
10			F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32		
11			F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32		
12			F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32		
13			F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32		
14			F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32		
15			F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32		
16			F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32		
17			F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32		
18		F73	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F73	
19		F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	
20	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32

ふんわりフワフワの風せんをつかまえるゲームです

2 不思議の国のバルーン

ゲームの遊び方

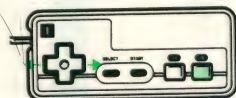
不思議の国の風せんクンが，外に逃げ出しちゃったんです。

この風せんは，不思議の国では，ごくごく普通の風せんクンなのに，外に出ると『ドラゴン』に変身しちゃいますっ！　不思議の国の魔力が消えて，元の姿に戻っちゃうのでたいへんなんです。

テレビ画面の上の方に飛んでるのが「風せんクン」，地面の上にいるのが「アリス」です。

「アリス」が上手に「風せんクン」を取ったところ。

「風せんクン」を不思議の国の外に逃がしちゃいましたあ！

まず，BGグラフィックを入れて，それからBASICプログラムを入力してネ。

それで，RUNすると，ゲームが始まります。地面の上をトコトコ歩いてるのは，みなさんが動かす『アリス』です。でね，**空をポコポコ飛んでる**のは，『風せんクン』。

『アリス』をジャンプさせて，『風せんクン』を取って！　『風せんクン』が右端までゆくまえに取らないと，だめなの…。

コントローラでアリスを左右に動かします。Aボタンで，ジャンプ！

配列F(I)，A(I)，B(I)，C(I)は，風せんの変数です。
A(I)，B(I)は風せんの位置です。

```
100  '●● GAME/02(チアキ+カナエ) ●●
110  DIM F(4),A(4),B(4),C(4)
120  VIEW:SPRITE ON:S=0:O=0
130  X=13:Y=17:VX=1:VY=0:M=3
150  FOR I=0 TO 13 STEP 2
160  FOR J=0 TO 26 STEP 2:COL
OR J,I,2:NEXT
170  NEXT
180  PALETS 2,13,53,37,18
250  LOCATE 0,23:PRINT" *SCOR
E= 0      HISCORE=";H;"    *";
260  FOR I=0 TO 27:COLOR I,21
,1:LOCATE I,22:PRINT"=";:NEX
T
300  FOR C=0 TO 30000
310  I=STRIG(0)
```

320行では，Aボタンを押してない時，女の子を左右に動かしてます。
330行では，Aボタンを押してる時，女の子をジャンプさせます。
390行から440行では，女の子を動かしてます。

```
320  IF I=0 THEN GOSUB 1000
330  IF I=8 THEN GOSUB 1100
350  R=X+VX:L=Y+VY
360  IF R<0 OR R>26 THEN VX=-
VX ←画面の端まで行ったら戻ります
370  IF L< 1 THEN VY=1
380  IF L>17 THEN VY=0
390  M=VX+VY+2-((VX+VX)<VY)*(
-VX-VX-VY-VY+4)
400  DEF MOVE(0)=SPRITE(1,M,1
,4,0,2)
410  POSITION 0,X*8+16,Y*8+24
420  X=X+VX
430  Y=Y+VY
440  MOVE 0 ←女の子を動かします
450  IF C MOD 5=0 GOSUB 1200
460  IF O THEN C=30000 ←ゲームオーバーなら…
470  NEXT
480  GOTO 1700
490  '
```

サブルーチン1では，女の子を左右に動かしてます。

```
1000  '●サブルーチン1●
1010  IF VY=1 THEN RETURN
1020  I=STICK(0)
1030  IF I=2 THEN VX=-1
1040  IF I=1 THEN VX= 1
1050  IF VY=-1 THEN VY=1 ←ジャンプするのをやめます
1060  RETURN
1070  '
```

サブルーチン2では，風せんを取ったかどうかを調べています。

```
1100  '●サブルーチン2●
1110  IF VY>0 OR VX=0 THEN RE
TURN
1120  VY=-1
1130  P$=SCR$(X-(VX=1),Y-1)
1140  Q$=SCR$(X-(VX=1),Y    )
1150  IF P$<>CHR$(238) AND Q$
=CHR$(238) THEN RETURN
1160  LOCATE X-(VX=1),Y-1:PRI
NT"*";:LOCATE X-(VX=1),Y:PRI
NT"*";
1170  RETURN
1180  '
```

サブルーチン3では，風せんの動きを管理してます。

```
1200  '●サブルーチン3●
1210  N=N+(N>0) ←もしN>0なら Nから1を引きます
1220  IF A(I)=-1 GOSUB 1300
1230  IF A(I)>-1 GOSUB 1400
1240  RETURN
```

```
1250  '
1300  '●サフ゛ルーチン4●
1310  IF N>0 OR RND(2)>0 THEN
 RETURN
1320  A(I)=0:B(I)=RND(10)+3:C
(I)=0:N=60-(S/50)*10
1330  RETURN
1340  '
1400  '●サフ゛ルーチン5●
1410  IF C(I)=0 THEN C(I)=RND
(7)-3
1420  IF SCR$(A(I),B(I)+1)="*
" THEN GOSUB 1600:RETURN
1430  LOCATE A(I),B(I):PRINT
" ";:LOCATE A(I),B(I)+1:PRIN
T" ";
1440  A(I)=A(I)+1
1450  B(I)=B(I)+SGN(C(I))
1460  IF A(i)=27 THEN A(I)=-1
:O=-1:RETURN
1470  IF B(I)=0 OR B(I)=14 B(
I)=B(I)-SGN(C(I)):C(I)=0
1480  COLOR A(I),B(I),2:LOCAT
E A(I),B(I):PRINT CHR$(207);
:LOCATE A(I),B(I)+1:PRINT CH
R$(238);
1490  RETURN
1500  '
1600  '●サフ゛ルーチン6●
1610  LOCATE A(I),B(I)   :PRIN
T" ";
1620  LOCATE A(I),B(I)+1:PRIN
T" ";
1630  LOCATE A(I),B(I)+2:PRIN
T" ";
1640  S=S+10:A(I)=-1
1650  LOCATE 8,23:PRINT S;
1660  RETURN
1670  '
1700  '●ケ゛ームオーハ゛ー●
1710  LOCATE 5,5:PRINT"サ゛ンネンテ゛
シタ!!"
1720  LOCATE 9,6:PRINT"ケ゛ームオー
ハ゛ーテ゛ス"
1730  INPUT A:GOTO 120
```

1320行: 風船の位置とかを決めてます

1460行: 風船が右まで行ったらー

1640行: 風船を表示するのをやめます

サブルーチン4では，風せんの出る位置を決めてます。

サブルーチン5では，風せんを動かして，表示してます。

サブルーチン6では，風せんを消してます。

●フローチャート

RUN
100－180▶変数・画面設定
250－260▶画面を作る
C＝0 TO 30000
310 ▶キー入力
Aボタンを押してない
サブルーチン1
Aボタンを押している
サブルーチン2
350－380 ▶これから動く所へ移って平気かを調べてます
390－440 ▶女のコを表示
C MOD 5＝0
サブルーチン3
0＝－1
ゲームオーバー
NEXT
サブルーチン1
1010－1050▶女のコ左右移動
RETURN
サブルーチン2
1110－1180▶女の子ジャンプ
RETURN
サブルーチン3
風船表示してない
サブルーチン4
風船表示してる
サブルーチン5
RETURN
サブルーチン4
1310－1320▶風船位置の設定
RETURN
サブルーチン5
1410 ▶風船の動き設定
風船が壊れる
サブルーチン6
サブルーチン6
1430 ▶風船を消す
1440－1470▶風船を移動
1480 ▶風船を表示
RETURN
サブルーチン6
1640－1650▶スコアを表示
RETURN

プログラムの説明

バックグランド面とスプライト面とを

同じように使う時は，なかなか大変なんです

このプログラムを作るのは，とても大変でした。もう，途中で作るのやめちゃおう！，なんて思ったくらい…。

作るの大変だったのは，**390行から440行**までの所です。ずっと，あーでもないこーでもない，て考えてやっとできました。

この部分は，スプライトをキャラクタみたいに扱えるようにしてるんです。

ファミリーコンピュータには，**バックグラウンド面とスプライト面**が，ありますね。バックグラウンドにはキャラクタ（文字）を表示して，スプライト面にはスプライト（アニメキャラクタ）を表示して…。この2つは，表示の仕方や動かし方，それに表示場所の決め方までゼンゼン違います。なので，両方とも使うとどうしてもゴチャゴチャになって，分からなくなっちゃいます。

どちらか一方の方法にできたら便利，と思いませんか？　そこで，スプライトの表示の仕方を，**キャラクタの表示の仕方**に合わせるように，作ってみたんです。

もし良ければ，次のプログラムを試してみてくださいね。このプログラムは，2×2文字のキャラクタをコントローラで動かします。

●プログラムを動かすとき
このサンプル・プログラムを動かすときに，注意して欲しいことがあります。表示するキャラクタが画面の端の方へ行き過ぎると，エラーが出てしまいます。なので，なるべく真ん中で動かすようにしましょう。

```
10  X=13:Y=10:VX=1
20  I=STICK(0)
30  IF  I=2  THEN  VX=-1
40  IF  I=1  THEN  VX= 1
50  LOCATE  X,Y:PRINT"   ";:LOC
ATE  X,Y+1:PRINT"   ";
60   X=X+VX:Y=Y+VY
70  LOCATE  X,Y:PRINT"⌒";:LOC
ATE  X,Y+1:PRINT"◡";
80  PAUSE  10:GOTO  20
```

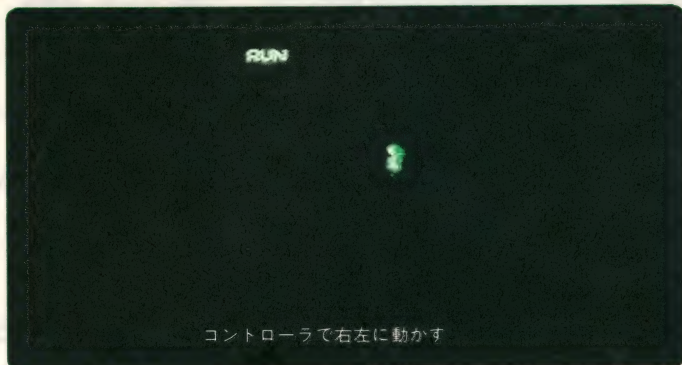
このサンプルプログラムは，大きい〈○〉を左右に動かすものです。

画面の真ん中らへんで，大きめの〈○〉が動きます。

今度は，次のプログラムを試してみてくださいね。**キャラクタを表示するようにスプライトを表示**するプログラム，なんです。

このサンプルプログラムはスプライトをキャラクタみたいに動かすものです。

```
10 SPRITE ON:X=13:Y=10:VX=1
20 I=STICK(Ø)
30 IF I=2 THEN VX=-1
40 IF I=1 THEN VX= 1
50 M=VX+VY+2-((VX+VX)<VY)*(-
VX-VX-VY-VY+4)
60 DEF MOVE(Ø)=SPRITE(1,M,1,
4,Ø,2)
70 POSITION Ø,X*8+16,Y*8+24
80  X=X+VX:Y=Y+VY
90 MØVE Ø:PAUSE 1Ø:GOTO 2Ø
```

画面の真ん中らへんで，スプライトが動きます。

どうです？　前のプログラムとそっくりだったとと思いません？

50行から90行が，スプライトを**キャラクタみたいに扱うように**，いろいろ仕事をしてくれてます。

50行では，キャラクタを動かすための方向の値が入ってる変数VX，VYを，スプライト用の変数Mに変えてます。この部分が，一番大切。一番大変…でした。この辺のことは，MOVE文のトコを見てくださいね。

60行では，変数Mの値をもとにして，女の子の動く向きを決めてます。70行では，動かし始める場所を決めてます。

●変数VX・VYと変数Mの関係

移動方向	VX	VY	M	移動方向	VX	VY	M
上	0	−1	1	下	0	1	5
右上	1	−1	2	左下	−1	1	6
右	1	0	3	左	−1	0	7
右下	1	1	4	左上	−1	−1	8

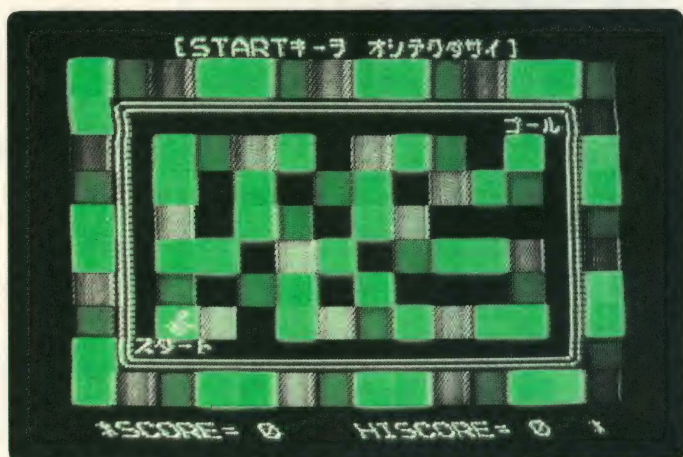
『すごろくの部屋』にとじこめられちゃったアリスちゃんを助けてネ

3 ドキドキすごろくルーレット

ゲームの遊び方

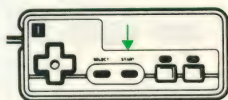
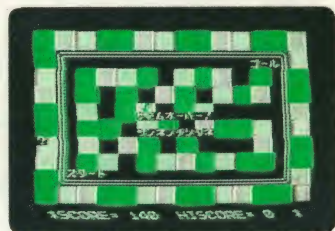
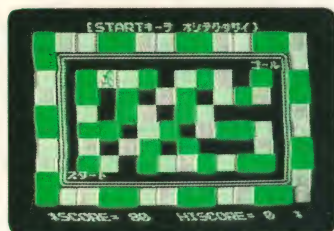
アリスは，すごろくの部屋にとじこめられちゃったの。ルーレットが止った色と同じ色の部屋にしか，アリスは移れません。ワーン，**おうちに帰りたいよー。**

アリスを助けてあげてネ。ねっ，おねがいっ！

テレビ画面の周りにあるのは，ルーレットです。真ん中にあるのは，すごろく。すごろくのスタート地点にいるのは，アリスです。

ゴールに近づくにつれて，ルーレットの回り方がはやくなります。もう，むつかしくって！

ルーレットの色が，アリスの周りにないからゲームオーバーになっちゃった。

まず，BGグラフィックを入れて，それからBASICプログラムを入れてネ。テレビ画面のまわりにあるのは，『ルーレット』。真ん中にあるのは，『すごろく』なの。それから，**すごろくのスタート地点**にいるのは，『アリス』です！

ゲームは，ルーレットをタイミング良く止めて，ゴールまでゆきます。進みたいマスの色とルーレットの色がおんなじなら，進めます。でも，ルーレットの色がアリスの周りにないと，だめなの…。

スタートボタンを１回，ポンって押すとルーレットが回ります。もう１回押すと止まります。

```
100  '●● GAME/03(チアキ+カナエ) ●●
110  DIM C(3),X(3),Y(3)
120  VIEW:SPRITE ON
130  C(0)=17:C(1)=21:C(2)=25:
C(3)=19
140  X(0)=0:X(1)=2:X(2)=0:X(3
)=-2
150  Y(0)=-2:Y(1)=0:Y(2)=2:Y(
3)=0
160  S=0:X=4:Y=15
190  PALSETS 2,13,53,37,18
200  FOR I=0 TO 3:PALSEB I,13
,48,48,C(I):NEXT
210  DEF SPRITE 0,(2,1,0,1,0)
="%$'&"←――――女の子の形にスプライトをする
220  SPRITE 0,X*8+16,Y*8+22
230  LOCATE 0,22:PRINT" *SCOR
E= 0      HISCORE=";H;"   *";
300  '●メインルーチン●
310  LOCATE 5,0:PRINT"[STARTキ
ーヲ オシテクダサイ]";
320  IF STRIG(0)=0 THEN 320
350  C=0:PAUSE 50
360  C=C+1:IF C=4 THEN C=0
370  PALETB C,13,48,48,C(C)
380  PALETB -(C+1)*(C<3),13,4
8,48,48
390  PAUSE-(7-(S/50))*(S<400)
400  IF STRIG(0)=0 THEN 360
410  LOCATE 5,0:PRINT"
                ";
420  C=C+1:IF C=4 THEN C=0
440  O=-1
450  FOR I=0 TO 3
460    P$=SCR$(X+X(I),Y+Y(I))
470    C$=SCR$(X+X(I),Y+Y(I),1
)
480    IF P$=CHR$(255) AND C=A
SC(C$) THEN GOSUB 1000
490  NEXT
500  IF O THEN 1100
510  PLAY"T1R0B#AA#GG"
520  PALETB C,13,48,48,C(C)
530  S=S+10
540  LOCATE 8,22:PRINT S;
550  GOTO 300
560  '
1000  '●サブルーチン1●
1010  O=0←――――エラーが出ないように
1020  X=X+X(I)
1030  Y=Y+Y(I)
1040  SPRITE 0,X*8+16,Y*8+22
1050  RETURN
1060  '
1100  '●ゲームオーバー●
1110  LOCATE 10,10:PRINT"ゲーム
ーバー!"
1120  LOCATE 10,12:PRINT"ザンネ
ンデシタ"
1130  INPUT A:GOTO 120
```

●配色番号とパレット

配色番号とパレットの関係について少々説明しましょう。

COLOR文やMOVE文やSPRITE文などで配色番号を指定することが出来ます。これは，色を指定するためですが，配色番号が直接に色を指定しているわけではありません。配色番号を指定した場合，その配色番号に入っているカラーコードが色を表すのであり，配色番号自身が色の値ではないのです。

ですから，同一の配色番号であっても入っているカラーコードが異なれば，指定する色も異なってきます。

この配色番号にカラーコードを入れる命令がPALET文であるわけです。香奈枝嬢は，これを利用してなかなか面白いゲームにしている。（小牧）

例

●配色番号＝17（青）

PALETB 0, 13, 48, 17, 13 : PRINT "□"

RETURN

●配色番号＝21（赤）

PALETB 0, 13, 48, 17, 13 : PRINT "□"

RETURN

（注）…□は

カナ + GRPH + [illegible]

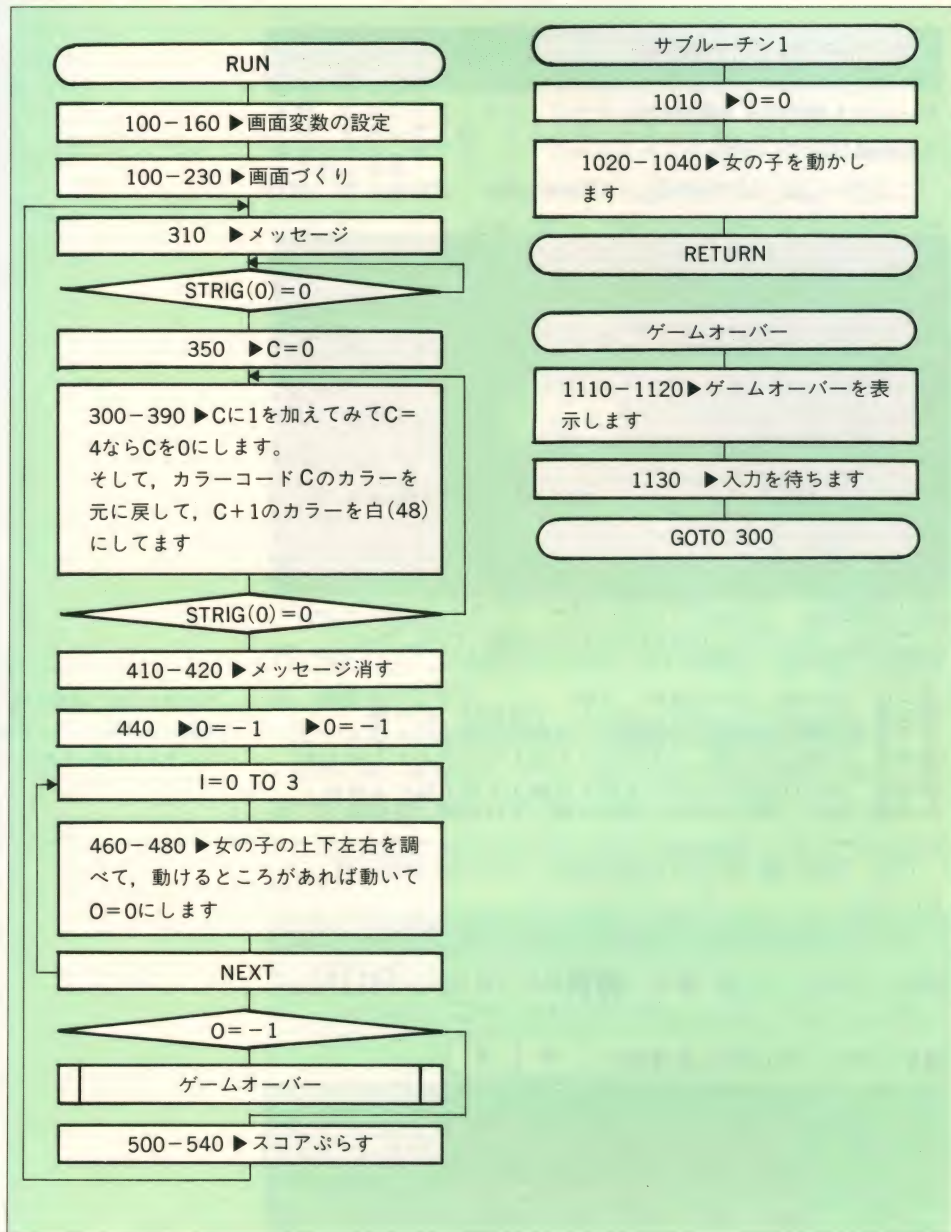
配色番号＝0→0

カラーコード→54

↓

白

●フローチャート

RUN
100－160 ▶画面変数の設定
100－230 ▶画面づくり
310 ▶メッセージ
STRIG(0)＝0
350 ▶C＝0
300－390 ▶Cに1を加えてみてC＝4ならCを0にします。
そして，カラーコードCのカラーを元に戻して，C＋1のカラーを白(48)にしてます
STRIG(0)＝0
410－420 ▶メッセージ消す
440 ▶O＝－1 ▶O＝－1
I＝0 TO 3
460－480 ▶女の子の上下左右を調べて，動けるところがあれば動いてO＝0にします
NEXT
O＝－1
ゲームオーバー
500－540 ▶スコアぷらす
サブルーチン1
1010 ▶O＝0
1020－1040▶女の子を動かします
RETURN
ゲームオーバー
1110－1120▶ゲームオーバーを表示します
1130 ▶入力を待ちます
GOTO 300

プログラムの説明

ルーレットをくるくる回すには,
こんな風にしています

このゲームは，なかなか楽しくできました…。うーん，やっぱりちーちゃんのアイデアが良かった，と思います。

それに，すごく難しい！　てとこがなくって助かりました。この前のゲームは，もう難しくって頭の中がゴチャゴチャになっちゃって…。こういう **頭の中がゴチャゴチャするようなゲーム**は，わたし，作るの苦手です。

その点このゲームは，わあーと楽しく作れました。嬉しいっ。

2つのちょっとしたテクニックを，使ってます。**くるくるルーレットみたい**に回るのと，**女の子が動けるか** を調べるとこ…です。

まず，くるくるルーレットの方から説明したい，と思います。ゲームを遊ぶと，くるくる回りますよね，ネオンサインみたいに…。

これは，プログラムの350行から400行が働いてるんです。

```
350  C=0:PAUSE 50
360  C=C+1:IF C=4 THEN C=0
370  PALETB C,13,48,48,C(C)
380  PALETB -(C+1)*(C<3),13,4
8,48,48
390  PAUSE-(7-(S/50))*(S<400)
400  IF STRIG(0)=0 THEN 360
```

プログラムのこの部分で，画面の色がルーレットみたいにクルクル回ってます。

でも，本当に働いてくれてるのかな？　ちょっと疑問に思ってしまいます。

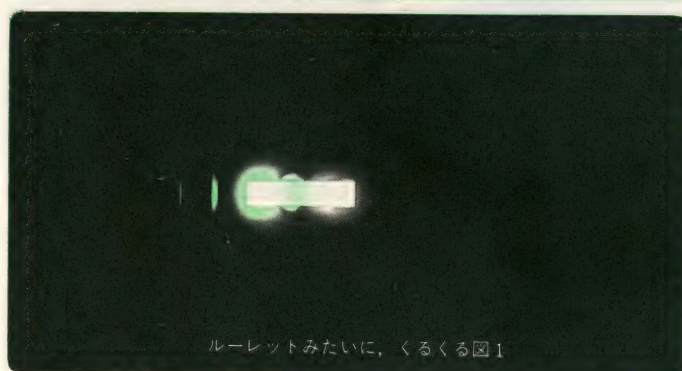
そこで，この部分をすこし簡単にしたサンプル・プログラムを，動かしてみたい…と思います。**背景**を作ってから，**RUN**してくださいね。

●サンプル・プログラムの背景

	0	1	2	3	4	5	6	7
0								
1	M70	M70	M71	M71	M72	M72	M73	M73
2	M70	M70	M71	M71	M72	M72	M73	M73

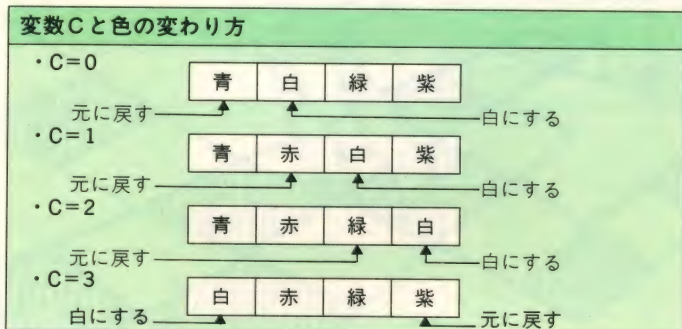
このサンプル・プログラムを動かすと，色がクルクル回ります。

```
10 DIM C(3):VIEW
20 C(0)=17:C(1)=21:C(2)=25:C
(3)=19
30 C=0
40 C=C+1:IF C=4 THEN C=0
50 PALETB C,13,48,48,C(C)
60 PALETB -(C+1)*(C<3),13,48
,48,48
70 PAUSE 10:GOTO 40
```

ルーレットみたいに，くるくる図1

ちゃんと動きました？　くるくる回してるのは，**40行から70行**が働いてくれてるおかげ…です。それ以前の **10行から30行**は，準備をしてます。

どうして，くるくる回るの？　**変数Cのおかげです。**

変数Cは，始めは0。でも，40行では1が足されて，0＋1＝1なりました。次は2。次は3。次は4…じゃなくて，0です。

このCの値に合わせて，白（カラーコード48）く光ってるわけなんですね。

今度は，女の子が動けるかを調べてるとこ，説明したいと思います。440行から500行が，働いてくれます。

でも，これじゃあ，なんだか分かりませんよね。なので，女の子が動けるかどうかを調べて，動けたら動いちゃうサンプル・プログラムを，作ってみました。

動かす前に，背景を描いておきましょう。

	10	11	12	13	14	16	17	18
9			M70	M70				
10			M70	M70				
11	M71	M71	M72	M72	M73	M73		
12	M71	M71	M72	M72	M73	M73		

配列X(1)，Y(1)には，女の子の周りを調べる位置が入ってます。
変数X，Yは，女の子の位置です。

```
100  DIM X(3),Y(3)
110  VIEW:SPRITE ON
120  X(Ø)=Ø:Y(1)=2:X(2)=Ø:X(3
)=-2
130  Y(Ø)=-2:Y(1)=Ø:Y(2)=2:Y(
3)=Ø
140  X=12:Y=11
150  DEF SPRITE Ø,(2,1,Ø,1,Ø)
="%$'&"
160  SPRITE Ø,X*8+16,Y*8+22
200  INPUT C ←移動したい配色番号を入力します
300  O=-1
310  FOR I=Ø TO 3      画面のキャラクタを調べてます
320   P$=SCR$(X+X(I),Y+Y(I))←
330   C$=SCR$(X+X(I),Y+Y(I),1
)
340   IF P$=CHR$(255) AND C=A
SC(C$) THEN GOSUB 1ØØØ
350  NEXT
360  IF O THEN END←O=-1なら終わります
370  GOTO 2ØØ
380  '
1ØØØ  O=Ø←これがクセ者なんですね
1Ø1Ø  X=X+X(I):Y=Y+Y(I)
1Ø2Ø  SPRITE Ø,X*8+16,Y*8+22
1Ø3Ø  RETURN
```

移動したい配色番号を入力しています

えい!

BGグラフィック画面03

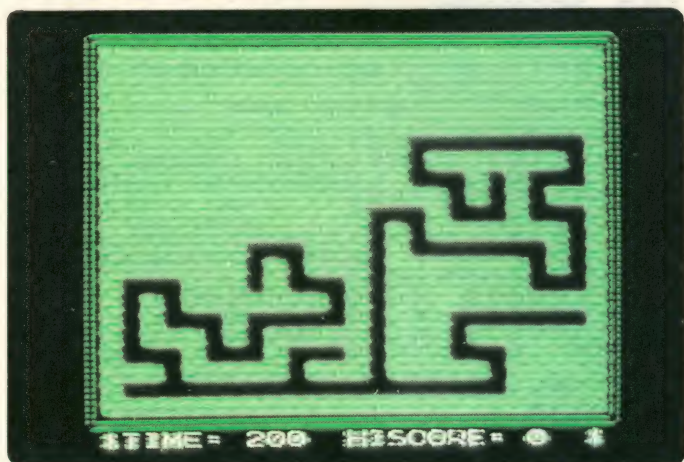
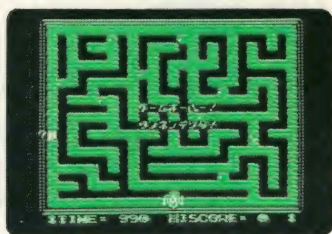
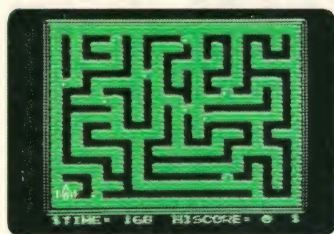
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
0																												
1	M70	M70	M71	M71	M72	M72	M73	M73	M70	M70	M71	M71	M72	M72	M73	M73	M70	M70	M71	M71	M72	M72	M73	M73	M70	M70	M71	M71
2	M70	M70	M71	M71	M72	M72	M73	M73	M70	M70	M71	M71	M72	M72	M73	M73	M70	M70	M71	M71	M72	M72	M73	M73	M70	M70	M71	M71
3	M73	M73	I60	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	I70	M72	M72
4	M73	M73	J20																				ゴ	ー	ル	J20	M72	M72
5	M72	M72	J20		M73	M73	M71	M71	M72	M72	M70	M70			M72	M72	M70	M70	M71	M71			M70	M70		J20	M73	M73
6	M72	M72	J20		M73	M73	M71	M71	M72	M72	M70	M70			M72	M72	M70	M70	M71	M71			M70	M70		J20	M73	M73
7	M71	M71	J20		M70	M70			M73	M73			M71	M71	M73	M73			M72	M72	M73	M73	M71	M71		J20	M70	M70
8	M71	M71	J20		M70	M70			M73	M73			M71	M71	M73	M73			M72	M72	M73	M73	M71	M71		J20	M70	M70
9	M70	M70	J20		M72	M72			M70	M70	M71	M71			M70	M70	M72	M72								J20	M71	M71
10	M70	M70	J20		M72	M72			M70	M70	M71	M71			M70	M70	M72	M72								J20	M71	M71
11	M73	M73	J20		M73	M73	M70	M70			M72	M72	M70	M70			M71	M71	M73	M73	M70	M70	M72	M72		J20	M72	M72
12	M73	M73	J20		M73	M73	M70	M70			M72	M72	M70	M70			M71	M71	M73	M73	M70	M70	M72	M72		J20	M72	M72
13	M72	M72	J20		M71	M71			M71	M71	M73	M73			M73	M73							M71	M71		J20	M73	M73
14	M72	M72	J20		M71	M71			M71	M71	M73	M73			M73	M73							M71	M71		J20	M73	M73
15	M71	M71	J20		M70	M70	M72	M72			M70	M70	M72	M72	M71	M71	M70	M70	M72	M72	M73	M73	M70	M70		J20	M70	M70
16	M71	M71	J20		M70	M70	M72	M72			M70	M70	M72	M72	M71	M71	M70	M70	M72	M72	M73	M73	M70	M70		J20	M70	M70
17	M70	M70	J20	ス	タ	ー	ト																			J20	M71	M71
18	M70	M70	J10	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J20	M71	M71
19	M73	M73	M72	M72	M71	M71	M70	M70	M73	M73	M72	M72	M71	M71	M70	M70	M73	M73	M72	M72	M71	M71	M70	M70	M73	M73	M72	M72
20	M73	M73	M72	M72	M71	M71	M70	M70	M73	M73	M72	M72	M71	M71	M70	M70	M73	M73	M72	M72	M71	M71	M70	M70	M73	M73	M72	M72

迷路の中にあるバクダンを回収しましょう！

4 メイズ・パニック

ゲームの遊び方

『コンピュータ・ルーム』に，**バクダン**が仕掛けられました。全部で10コ…。はやく回収しないと，**日本じゅうがパニック**になっちゃう。でも，このコンピュータ・ルームに入れるのは，わいいっ『スペースシップ』だけなんです。ワァ！　困っちゃうー。

テレビ画面には，迷路が作られます。画面の右下にあるスペースシープを動かして，バクダンを回収します。

迷路のカベにぶつかったり，タイムが0になっちゃうとゲームオーバーです。

まず，BGグラフィックを入れて。それから，BASICプログラムを入力してネ。RUNすると，どんどん迷路(= メイズ)ができてくるの。で，できた迷路の中に『バクダン』が10コ，**仕掛けられるんです。**

コントローラのAボタンを押したりして，スペースシップを動かします。

このバクダン10コを，バクダン処理船に乗って回収するゲームです。処理船は，クルクル回ってるから動きたい方向の時，Aボタンを押してね。壁にぶつかったり，タイムが0になるとゲームオーバー。

```
100  '●● GAME/04(チアキ+カナエ) ●●
110  DIM X(3),Y(3),M(3),N(3),
A(8),B(8)
120  CGSET 1:VIEW:SPRITE ON
130  X(0)=0:X(1)=2:X(2)=0:X(3
)=-2:Y(0)=-2:Y(1)=0:Y(2)=2:Y
(3)=0
140  M(0)=0:M(1)=1:M(2)=0:M(3
)=-1:N(0)=-1:N(1)=0:N(2)=1:N
(3)=0
150  RESTORE
160  FOR I-1 TO 8:READ A(I),B
(I):NEXT
170  DATA 0,-1,1,-1,1,0,1,1,0
,1,-1,1,-1,0,-1,-1
180  PLAY"T1Y2M1V15R0"
190  '
200  LOCATE 0,21
210  PRINT CHR$(226);:FOR I=1
 TO 25:PRINT CHR$(195);:NEXT
:PRINT CHR$(226);
220  LOCATE 0,22
230  PRINT CHR$(224);:FOR I=1
 TO 25:PRINT CHR$(227);:NEXT
:PRINT CHR$(225);
260  LOCATE 2,20:PRINT" ";
270  PALETB 0,13,48,22,2
280  LOCATE 0,23:PRINT" *TIME
= 500  HISCORE=";H;"  *";
290  '
300  '●メイロツクリ●
310  C=0
320  FOR Y=20 TO 2 STEP -2
330  FOR X=2 TO 24 STEP 2
340    IF SCR$(X,Y)=" " GOSUB
1200:IF O=0 GOSUB 1000
350  NEXT
360  NEXT
370  IF C<119 THEN 320
380  GOSUB 1300:X=2:Y=20:M=1:
V=4:T=500:SPRITE ON
390  '
400  '●メインルーチン●
410  I=STRIG(0):V=4
420  IF I=0 M=1-(M+1)*(M
<7):V=1
430  P$=SCR$(X+A(M),Y+B(M))
440  IF P$<>" " AND V=4 GOSUB
 1400
450  DEF MOVE(0)=SPRITE(9,M,1
,V,0,0)
460  POSITION 0,X*8+12,Y*8+18
470  IF V=4 X=X+A(M):Y=Y+B(M)
490  MOVE 0:PAUSE 15
500  T=T-1:IF T<0 GOTO 1600
510  LOCATE 7,23:PRINT T;
520  GOTO 400
530  '
1000  '●サブルーチン1●
1010  O=-1
1020  FOR I=1 TO 10
1030    R=RND(4)
```

配列X(I), Y(I)には, 迷路を作るためのデータが入ってます。
配列M(I), N(I)には, スペースシップを動かすためのデータが入ってます。

ここでは, 迷路を作ってます。

メインルーチンでは, スペースシップを動かしてます。

サブルーチン1では, 迷路を1コ作ってます。

```
1040  IF SCR$(M+X(R),N+Y(R))
=CHR$(195) I=10:O=0
1050 NEXT
1060 IF O RETURN
1070 LOCATE M+X(R),N+Y(R):PR
INT" ";
1080 LOCATE M+M(R),N+N(R):PR
INT" ";
1090 M=M+X(R)
1100 N=N+Y(R)
1110 C=C+1
1120 GOTO 1000
1130 '
1200 '●サブルーチン2●
1210 O=-1
1220 FOR I=0 TO 3
1230  IF SCR$(X+X(I),Y+Y(I))
=CHR$(195) O=0
1240 NEXT
1250 IF O RETURN
1260 M=X:N=Y
1270 RETURN
1280 '
1300 '●サブルーチン3●
1310 FOR I=1 TO 10
1320  X=RND(23)+2
1330  Y=RND(19)+2
1340  IF SCR$(X,Y)<>" " GOTO
 1320
1350  LOCATE X,Y:PRINT CHR$(
215);
1360 NEXT
1370 RETURN
1380 '
1400 '●サブルーチン4●
1410 IF P$=CHR$(195) T=0
1420 IF P$=CHR$(215) S=S+1:L
OCATE X+A(M),Y+B(M):PRINT" "
;:PLAY"O5C0BCB"
1430 IF P$=CHR$(199) GOSUB 1500
1440 RETURN
1450 '
1500 '●サブルーチン5●
1510 IF S<100 T=0:RETURN
1520 LOCATE 10,10:PRINT"オメデ
トウ!!!";
1530 LOCATE 10,12:PRINT"セイコウ
シマシタ";
1540 INPUT A:GOTO 120
1550 '
1600 '●ゲームオーバー●
1610 LOCATE 10,10:PRINT"ゲーム
オーバー!";
1620 LOCATE 10,12:PRINT"ザンネ
ンデシタ!";
1630 INPUT A:GOTO 120
1640 '
```

迷路の進む方向（1090〜1100）

サブルーチン2では，スペースシップが動けるかどうかを調べてます。

サブルーチン3では，バクダンを表示してます。

バクダンの位置（1320〜1330）

サブルーチン4では，スペースシップが何にぶつかったのかを調べてます。

プログラムの説明

迷路の作り方を

考えてみました…

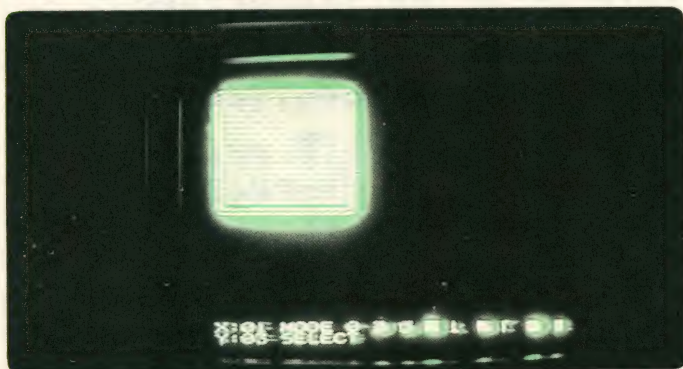
このプログラムで，とても大切な部分……。それは，**迷路を作る仕事をしている所**です。

プログラムを作る時，いっぺんに全部考えよう！　とするのは良くないみたい。なるべく，細かく分けて，**少しずつ**考えるようにすると，いいんです。

では，迷路を作るプログラム，の作り方を説明したいと思います。

大きい迷路は，考えるのたいへんです。なので，すこし小さめの迷路にしてみました…。大きさは11×11です。

下の図を見て，**BGグラフィック**を使って**背景**を描いておいてくださいネ。

	0	1	2	3	4	5	6	7	8	9	10
0	I60	J30	J30	J30	J30	J30	J30	J30	J30	J30	I 70
1	J20	F30	F30	F30	F30	F30	F30	F30	F30	F30	J70
2	J20	F30		F30	F30	F30	F30	F30	F30	F30	J20
3	J20	F30	F30	F30	F30	F30	F30	F30	F30	F30	J20
4	J20	F30	F30	F30	F30	F30	F30	F30	F30	F30	J20
5	J20	F30	F30	F30	F30	F30	F30	F30	F30	F30	J20
6	J20	F30	F30	F30	F30	F30	F30	F30	F30	F30	J20
7	J20	F30	F30	F30	F30	F30	F30	F30	F30	F30	J20
8	J20	F30	F30	F30	F30	F30	F30	F30	F30	F30	J20
9	J20	F30	F30	F30	F30	F30	F30	F30	F30	F30	J20
10	J00	J30	J30	J30	J30	J30	J30	J30	J30	J30	J10

●変数の説明

X，Y→迷路を作りはじめる位置

N，M→今作ってる迷路の位置

X（I），Y（I）→迷路が上下左右に動く時の変化する座標の値

O→迷路が行き止ったかどうかの値

初めに，迷路を一本だけ作るプログラムを，作ってみますね。

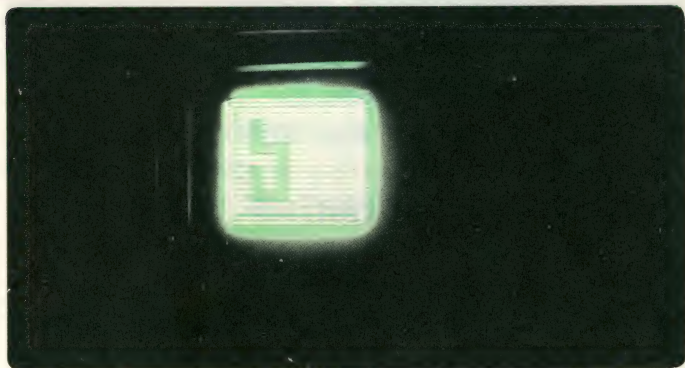
迷路は，たくさんの道で，できてます。でも…，いっぺんにたくさん作るのは，難しいと思います。**一本だけなら**，そんなに難しくないですよネ。

このプログラムをキー入力して，RUN命令で動かすと，ちゃんと迷路が一本だけ，できるんです。なかなか，エライでしょ。

外へ出ちゃわないように。それから，迷路どうしくっつかないように…。動けなくなるまで，迷路を作ります。

300から400行では，進んでく迷路の先が《■》じゃなかったらヤメます。

```
100  DIM X(3),Y(3),M(3),N(3)
110  VIEW
120  X=2:Y=2
130  X(0)=0:X(1)=2:X(2)=0:X(3
)=-2:Y(0)=-2:Y(1)=0:Y(2)=2:Y
(3)=0
140  M(0)=0:M(1)=1:M(2)=0:M(3
)=-1:N(0)=-1:N(1)=0:N(2)=1:N
(3)=0
200  M=X:N=Y:GOSUB 300
210  END
220  '
300  O=-1
310  FOR I=1 TO 10
320   R=RND(4)
330   IF SCR$(M+X(R),N+Y(R))=
CHR$(195) THEN I=10:O=0
340  NEXT
350  IF O THEN RETURN
360  LOCATE M+X(R),N+Y(R):PRI
NT"  ";
370  LOCATE M+M(R),N+N(R):PRI
NT"  ";
380  M=M+X(R)
390  N=N+Y(R)
400  GOTO 300
410  '
```

ねー。やっぱり迷路が1本できました

```
100 DIM X(3),Y(3),M(3),N(3)
110 VIEW
120 X=2:Y=2
130 X(0)=0:X(1)=2:X(2)=0:X(3
)=-2:Y(0)=-2:Y(1)=0:Y(2)=2:Y
(3)=0
140 M(0)=0:M(1)=1:M(2)=0:M(3
)=-1:N(0)=-1:N(1)=0:N(2)=1:N
(3)=0
200 FOR X=2 TO 8 STEP 2
210 FOR Y=2 TO 8 STEP 2
220   IF SCR$(X,Y)=" " THEN M
=X:N=Y:GOSUB 300
230 NEXT
240 NEXT
250 END
260 '
300 O=-1
310 FOR I=1 TO 10
320   R=RND(4)
330   IF SCR$(M+X(R),N+Y(R))=
CHR$(195) THEN I=10:O=0
340 NEXT
350 IF O THEN RETURN
360 LOCATE M+X(R),N+Y(R):PRI
NT" ";
370 LOCATE M+M(R),N+N(R):PRI
NT" ";
380 M=M+X(R)
390 N=N+Y(R)
400 GOTO 300
410 '
```

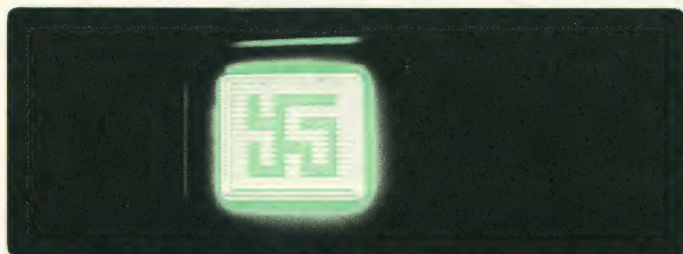
●110−110
配列宣言と画面の設定をしています。

●120−140
変数の値を決めてます。

●200−240
迷路を作り始める位置を決めてます。作り始める位置がスペース（" "）じゃない時は，作り始めません。

●300−350
迷路を作る方向を決めてます。行き止まりなら戻ります。

●360−370
迷路を作ります。

●380−390
迷路の座標を移動します。

ちゃあんと迷路が，できました。

さっきの一本道の迷路から，新しく迷路ができてゆきます。

このプログラムで，みなさんにちょっとだけ考えてほしいな，て思ったことあります…それは，**220行**の**IF SCR$(X, Y) = " " THEN**のこと。

どうしてこんな命令があると思います？

もしも，時間があったなら，考えてみてくださいね。これがないと，上手に迷路ができないんです。

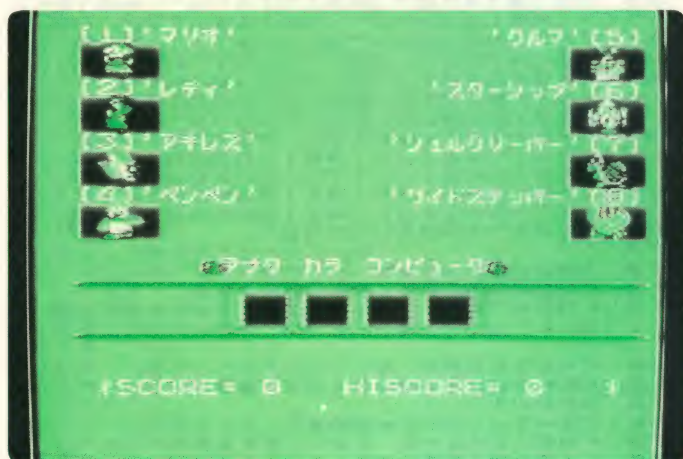
あなたの記憶力を調べてしまいます

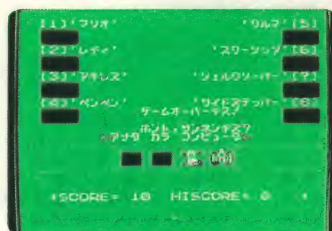
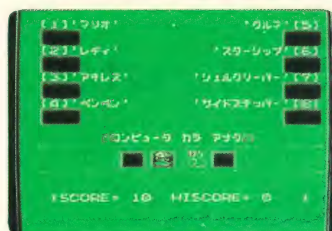
5 あなたの能力，どれくらい?

ゲームの遊び方

コンピュータと，**記憶力**を競いますー。

コンピュータの出すマークを，**どのくらい覚えられる?**，というわけなんです。なんとなく，カンタンそう…。ううん，でもぉ，数が多くなるとわけが分らなくなっちゃうもん。

テレビ画面の上の方には，コンピュータの選ぶマーク8種類が描いてあります。画面の真ん中よりちょっと下には，マークが出てくる所があります。

コンピュータがマークを選んで，出題してきました。

さーて，答えられるかな？

1 2 3 4 5 6 7 8

マークの番号を，キーボードの数字のキーで答えてネ

まず，BGグラフィックを入れて，それから，**BASICプログラム**を入力してネっ。

このゲームでは，コンピュータと記憶力を競うんです。コンピュータが，8種類のマークから選んで出します。なので，そのとおりにマークの番号をキーボードから入力してくださいっ！　でも間違えちゃうし，ざんねん，**ゲームオーバー**です。

20コくらいになると，もう，憶えられなくなっちゃうもん。

```
100  '●● GAME/05(チアキ+カナエ) ●●
110  DIM A(3),M$(7)
120  VIEW:CGSET1,0:SPRITE ON
130  PALETB 0,1,48,13,22
140  PALETS 0,1,48,22,1
150  PALETS 1,13,48,38,18
160  S=0:D=0:D$=""
170  A(0)=152:A(1)=128:A(2)=1
4:A(3)=80
180  M$(0)="C":M$(1)="D":M$(2
)="E":M$(3)="F":M$(4)="G":M$
(5)="A":M$(6)="B":M$(7)="A#"
190  LOCATE21,20:PRINT H;
200  I=0:C=0:J=0:GOSUB1400
210  I=1:C=2:J=28:GOSUB1400
220  I=2:C=0:J=68:GOSUB1400
230  I=3:C=1:J=104:GOSUB1400
240  I=4:C=0:J=136:GOSUB1400
250  I=5:C=0:J=172:GOSUB1400
260  I=6:C=3:J=184:GOSUB1400
270  I=7:C=1:J=192:GOSUB1400
280  PLAY"T303V15R0"
290  '
300  '●メインルーチン●
310  GOSUB1300:PAUSE130
410  LOCATE6,13:PRINT CHR$(20
7);"コンピュータ カラ アナタ";CHR$(207
);
420  PAUSE200
440  D$=CHR$(RND(8)+49)+D$
450  D=D+1
460  C$="    "+D$+"    "
470  FOR I=4+D TO1 STEP-1
480   I$=MID$(C$,I,4)
490   GOSUB1000:PAUSE50
500  NEXT
510  '
550  GOSUB1300
560  LOCATE 6,13:PRINT CHR$(2
15);"アナタ カラ コンピュータ";CHR$(21
15);
600  O=0:C$=""
610  FOR I=1 TO D
620   GOSUB1200
630   C$=I$+C$
640   I$=LEFT$(C$+"    ",4)
650   GOSUB1000: PAUSE50
660  NEXT
670  IF D$<>C$ THEN 1500
700  FOR I=1TO3
710  C$=" "+C$:I$=LEFT$(C$+"
  ",4)
720  GOSUB1000:PAUSE50
730  NEXT
750  S=S+10
760  LOCATE8,20:PRINT S;
770  GOTO300
780  '
1000  '●サブルーチン1●
1010  FORJ=0TO7:SPRITE J:NEXT
1040  FOR J=0TO3
```

スプライトを決めてます（200行～270行）

スコアに10点をプラス（750行）

配列M$には，音のデータが入ってます。200行から270行では，スプライトの定義をしてます。

440行から500行では，コンピュータが問題をだしてます。I$に表示したいスプライト4種類を入れて，1000行にあるサブルーチンを呼ぶとそのスプライトが表示されます。

600行から730行では，みなさんが問題に答えるのをしてます。600行から660行で入力した数をスプライトにして画面に表示してます。700行から730行で，表示したスプライトを左端へ順番に消してます。

```
1050   J$=MID$(I$,J+1,1)
1070   IF J$<>" " THEN SPRITE
 VAL(J$)-1,A(J),142:PLAY M$(
VAL(J$)-1)
1080  NEXT
1090  RETURN
1100  '
1200  '●サブルーチン2●
1210  I$=INKEY$
1220  IF I$<"1"ORI$>"8" GOTO1
200←――――キー入力待ちしてます
1230  RETURN
1240  '
1300  '●サブルーチン3●
1310  C=0
1320  FOR X=1TO25 STEP24
1330  FOR Y=1TO10 STEP3
1340   SPRITE C,X*8+16,Y*8+22
1350   C=C+1
1360  NEXT
1370  NEXT
1380  RETURN
1390  '
1400  '●サブルーチン4●
1410  DEF SPRITE I,(C,1,0,0,0
)=CHR$(J)+CHR$(J+1)+CHR$(J+2
)+CHR$(J+3)
1420  RETURN
1430  '
1500  '●ゲームオーバー●
1510  LOCATE 10,14:PRINT"ゲーム
オーバー";
1520  LOCATE 10,16:PRINT"ホント,
ザンネンデス";
1530  INPUT A$:IF S>H H=S
1540  GOTO160
```

サブルーチン３では，メニューみたいに画面にスプライトを表示してます。270行よりあとのプログラムならここを呼べば，画面にスプライトを表示できるんです。

●サンプルプログラム

RUN
100−150 ▶画面の設定＋α
160−280 ▶変数の設定とスプライトの設定をしてます
サブルーチン3
410−420 ▶メッセージ表示
440−450 ▶出すカタチ設定
I＝4＋D TO 1
480 4文字取出す
サブルーチン1
NEXT
サブルーチン3
500−570 ▶メッセージ＋α
I＝1 TO D
サブルーチン2
630−640 ▶4文字を取り出す
サブルーチン1
NEXT
間違って入力
ゲームオーバー
700−730 ▶4つ表示されてるスプライトを左側へ消します
750−760 ▶スコア加算表示
サブルーチン1
1010 ▶スプライト消す
J＝0 TO 3
1050 ▶1文字を取り出す
J$<>" "
1070 ▶スプライト表示
NEXT
RETURN
サブルーチン2
1210−1220▶1文字分だけ入力
RETURN
サブルーチン3
X＝1 TO 25 STEP 24
Y＝1 TO 10 STEP 2
1340−1350▶スプライト表示
NEXT
NEXT
RETURN
ゲームオーバー
1510−1530▶ゲームオーバーの表示とハイスコアの設定。それからキー入力待ちをします
GOTO 160

プログラムの説明

OM ERRORが出ると，
困っちゃう…

このプログラムは，メモリいっぱい，なんです。なので，プログラムを動かしてると，もしかすると「**OM　ERROR**」になっちゃうかも…。メモリ不足なんです。

そんな時は，次のようにしてください…ね。REM文を，スッキリさせます。

●OM　ERROR
メモリ量が不足すると出るエラーメーセージです。プログラムを少なくして，メモリを増やしましょう。

REM文のメッセージをなくして，メモリを多くします。でも，REM文もなくしちゃう，なんてことはしないでくださいね。ERRORが出るかもしれないんです…。

こんなふうに，**OM　ERROR**が出ちゃうこと，みなさんもあると思います。どうです？

ファミリーベーシックって**メモリ容量が2Kだから**，どうしてもメモリ不足になりがち…なんですね。

こんなときは，次のようにすると良いみたい…です。

①REM文は，なるべく少なくします。

REM文は，とてもたくさんのメモリを使います。メモリ不足のときは，なるべく無くしてしまいましょう…。

使うメモリ数	プログラム
17バイト	1000 '■REMステートメント■
5バイト	1000 '

●バイト（byte）
バイトとは，マイコンが扱う単位のことです。マイコンが一度に扱える数やデータのことです。そうですね，マイコンにとって1バイト＝1コ。10バイト＝10コというようなものなんです。

②空白(スペース)を少なくします

空白〈　〉は，1つ＝1バイトです。メモリ不足のときは，空白をなるべく少なくしましょう。でも，あんまり少ないの，困ります。空白があんまり少ないと，プログラムがゴチャゴチャ見にくくなっちゃいます…。ほどほどに，ネ。

```
10  FOR□X=0□TO□27
20  □FOR□Y=0□TO□22
30  □□LOCATE□X,Y:PRINT□CHR$(2
15);
40  □NEXT
50  NEXT
60  GOTO□60
■
```

空白がかなりある

●メモリ＝76バイト
こんなふうに，キレイなプログラムだと76バイトのメモリを使います。

```
10  FORX=0TO27
20  FORY=0TO22
30  LOCATEX,Y:PRINTCHR$(215);
40  NEXT
50  NEXT
60  GOTO60
■
```

空白を取りました。

●メモリ＝63バイト
空白を取っちゃうと，13バイト少なくてすみます。でも，ちょっぴり見にくくなりました。

まだまだ他にも，**メモリーを少なくする方法**があるんです。残りは，次のゲームのとこで，説明したいと思います。

BGグラフィック画面05

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
0	[	1	]	'	マ	リ	オ	'													'	ク	ル	マ	'	[	5	]
1	M60	M60	M60	M60																					M60	M60	M60	M60
2	M60	M60	M60	M60																					M60	M60	M60	M60
3	[	2	]	'	レ	デ	ィ	'										'	ス	タ	ー	シ	ッ	プ	'	[	6	]
4	M60	M60	M60	M60																					M60	M60	M60	M60
5	M60	M60	M60	M60																					M60	M60	M60	M60
6	[	3	]	'	ア	キ	レ	ス	'							'	シ	ェ	ル	ク	リ	ー	パ	ー	'	[	7	]
7	M60	M60	M60	M60																					M60	M60	M60	M60
8	M60	M60	M60	M60																					M60	M60	M60	M60
9	[		]	'	ペ	ン	ペ	ン	'							'	サ	イ	ド	ス	テ	ッ	パ	ー	'	[	8	]
10	M60	M60	M60	M60																					M60	M60	M60	M60
11	M60	M60	M60	M60																					M60	M60	M60	M60
12																												
13																												
14	K50	K50	K50	K50	K50	K50	K50	K20	K50	K50	K20	K50	K50	K20	K50	K50	K20	K50	K50	K20	K50	K50	K50	K50	K50	K50	K50	K50
15								K60	M60	M60	K60	M60	M60	K60	M60	M60	K60	M60	M60	K60								
16								K60	M60	M60	K60	M60	M60	K60	M60	M60	K60	M60	M60	K60								
17	K50	K50	K50	K50	K50	K50	K50	K10	K50	K50	K10	K50	K50	K10	K50	K50	K10	K50	K50	K10	K50	K50	K50	K50	K50	K50	K50	K50
18																												
19																												
20		*	S	C	O	R	E	=		0				H	I	S	C	O	R	E	=		0				*	

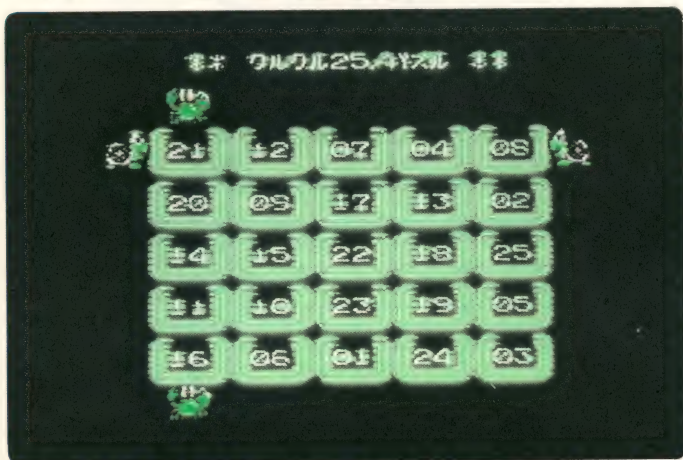
バラバラになってる数字を元通りに戻しましょ

6 くるくる25パネル

ゲームの遊び方

はじめは，順番に並んでるの。

でもね，それをコンピュータが**バラバラに**しちゃうんです。このバラバラになっちゃった『数』を元通りに戻してください。

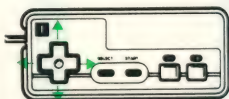
時間は無制限。**じーっくり考え**て，クルクルしてネ。

テレビ画面には，1から25までの数字が描いてありますね。コレを，元のとおり順番に並べるんでーす。

まず，BGグラフィックを入れて…。それから，BASICプログラムを入力してネ。

このゲームは，バラバラにきれされちた1から25までの数を，**元のとおりの順番に並べるんです。**

順番に並べるには，初めに回転させたいナっていう数の列を決めます。タテの列を回転させたい時はカニさんを，ヨコの時はカメさんをその列に動かすの。で，Aボタンをポンと押してから，コントローラを回転させたい方向に押してね。

それで，またAボタンを押すと，カニさんやカメさんを動かせるんですっ。みなさん，だいたい分ってくれましたあ？

時間は無制限だから，じーっくり考えて遊んでネ。でも，それでもむつかしいヨって思ったら，310行の〈TO〉の後の数を少なくしてみてねー。

コントローラで，回転させたい数の列を決めたり，向きを決めて回転させたりします。
Aボタンで，カニやカメを動かすのか，数の列を動かすのかを切り替えます。
数が順番に並んだら，セレクトキーを押しまくってネ。

```
100  '●● GAME/06(チアキ+カナエ) ●●
110  DIM D(4,4),S(4)
120  CGSET1,2:VIEW:SPRITE ON
160  I=1
170  FOR Y=0TO4
180  FOR X=0TO4
190  D(X,Y)=I:I=I+1  ←パズルのコマの値を決めてます
200  NEXT
210  NEXT
220  DEF SPRITE 0,(3,1,0,1,0)
=CHR$(185)+CHR$(184)+CHR$(18
7)+CHR$(186)
230  DEF SPRITE 1,(3,1,0,0,0)
=CHR$(184)+CHR$(185)+CHR$(18
6)+CHR$(187)
240  DEF SPRITE 2,(2,1,0,0,0)
=CHR$(192)+CHR$(193)+CHR$(19
4)+CHR$(195)
250  DEF SPRITE 3,(2,1,0,0,0)
=CHR$(192)+CHR$(193)+CHR$(19
4)+CHR$(195)
290  '
300  '●メインルーチン●
310  FOR N=1TO10
320  X=RND(5):Y=RND(5)
330  I=RND(2):VX=0:VY=0
340  IF I=0 VX=RND(2)*2-1
350  IF I=1 VY=RND(2)*2-1
360  FOR C=1 TO RND(4)+1
370  GOSUB1000  ←コマを動かします
380  NEXT
390  NEXT
400  X=0:Y=0
410  '
420  GOSUB1700
430  IF X+VX=-1 OR X+VX=5 VX=0

440  IF Y=VY=-1 OR Y+VY=5 VY=0
450  X=X+VX
460  Y=Y+VY
470  SPRITE 2,56+X*32, 38
480  SPRITE 3,56+X*32,174
490  SPRITE 0, 32,58+Y*24
500  SPRITE 1,208,58+Y*24
510  IF STRIG(0)=2 GOTO1800
520  IF STRIG(0)<>8 GOTO420
530  PLAY"#A1"
540  '
550  GOSUB1700
560  IF VX+VY<>0 GOSUB1000  ←コマを動かします
570  IF STRIG(0)=2 GOTO1800
580  IF STRIG(0)<>8 GOTO550
590  PLAY"#G1"
600  GOTO420
610  '
1000  '●サブルーチン1●
1010  IF VX<>0 GOSUB1500:GOSU
B1100:GOSUB1300
1020  IF VY<>0 GOSUB1600:GOSU
B1200:GOSUB1400
1030  RETURN
1040  '
```

```
1100  '●サブルーチン2●
1120  FOR  I=0TO4
1130  J=I+VX
1140  IF  J=-1  J=4      ←ヨコ向きに回転
1150  IF  J= 5  J=0
1160  D(J,Y)=S(I)
1170  NEXT
1180  RETURN
1190  '
1200  '●サブルーチン3●
1220  FOR  I=0TO4
1230  J=I+VY
1240  IF  J=-1  J=4      ←タテ向きに回転
1250  IF  J= 5  J=0
1260  D(X,J)=S(I)
1270  NEXT
1280  RETURN
1290  '
1300  '●サブルーチン4●
1310  FOR  I=0TO4
1320  I$=STR$(D(I,Y))
1330  LOCATE  5+I*4,5+Y*3
1340  PRINT  RIGHT$("0"+RIGHT$
(I$,LEN(I$)-1),2);
1350  NEXT
1360  RETURN
1370  '
1400  '●サブルーチン5●
1410  FOR  I=0TO4
1420  I$=STR$(D(X,I))
1430  LOCATE  5+X*4,5+I*3
1440  PRINT  RIGHT$("0"+RIGHT$
(I$,LEN(I$)-1),2);
1450  NEXT
1460  RETURN
1470  '
1500  '●サブルーチン6●
1510  FOR  I=0TO4
1520  S(I)=D(I,Y)
1530  NEXT
1540  RETURN
1550  '
1600  '●サブルーチン7●
1610  FOR  I=0TO4
1620  S(I)=D(X,I)
1630  NEXT
1640  RETURN
1650  '
1700  '●サブルーチン8●
1710  I=STICK(0):VX=0:VY=0
1720  IF  I=2  VX=-1
1730  IF  I=1  VX= 1     ←コントローラ入力
1740  IF  I=8  VY=-1
1750  IF  I=4  VY= 1
1760  RETURN
1770  '
1800  '●ゲームオーバー●
1810  O=-1:I=1:PLAY"#F1"
1820  FOR  Y=0TO4
1830  FOR  X=0TO4
1840  IF  D(X,Y)<>I  O=0←違ってるから，O=012…
```

サブルーチン1は，変数VXとVYの値によって右向きに回したり左向きに回したり下向きに回したり上向きに回したりしてます。回す場所は，変数X，Yで決まります。サブルーチン2，3，4，5，6，7は，このサブルーチン1の子分みたいなものです。

サブルーチン8は，みなさんが操作したコントローラの動きを，変数に入れるものです。

```
1850  I=I+1
1860  NEXT
1870  NEXT
1880  IF O GOTO2000
1890  LOCATE 9,10:PRINT"GAMEO
VERデス";
1900  INPUT"",A
1910  GOTO130
1920  '
2000  '●サブルーチン7●
2010  LOCATE10,10:PRINT"YOU W
IN!!!";
2020  INPUT"",A
2030  GOTO130
```

←正しければ，2000行へ

プログラムの説明

メモリを上手に使うには，
こんな風にします

このプログラムも，またメモリいっぱいになってしまいました。わたし，どうしても一生懸命に作っちゃう…みたいです。

一応，ちゃんと動きます。でも，もしも動かしている途中で **OM ERROR**が出たら，REM文のメッセージを無くしてみてくださいね。そうすれば，ちゃあんと動くはずです。

```
100  '●● GAME/06(チアキ+カナエ) ●●
```

↓

```
100  '
```

さて，この前のつづき。OM ERROR対策のし方…です。

③マルチステートメントをたくさん使う。

マルチステートメントは，1つの行にたくさんの命令を入れること，なんだそうです。ファミリーベーシックでは，マルチステートメントを使えば，1行に255文字まで入れられる，そうです…。

これを使って行数が少なくなると，減った行数×3バイト，メモリの節約になります。

●マルチステートメント
1つの行に，いくつも命令や文を入れることをマルチステートメントといいます。命令と命令の間にはコロン（:）をはさんでおきます。

```
10  FOR  X=0TO27
20  FOR  Y=0TO21
30  LOCATE  X,Y:PRINT  CHR$(215
);
40  NEXT
50  NEXT
60  END
```

```
10  FOR  X=0TO27:FOR  Y=0TO21:L
OCATE  X,Y:PRINT  CHR$(215);:N
EXT:NEXT:END
```

上のプログラムは，メモリを64バイト使っています。このプログラムをマルチステートメントを利用して1行にしてみました。使うメモリは49バイトになりました。でも，その分プログラムが見にくくなっちゃいましたね。

④IF文をかんたんに…

みなさんは，もう気づいちゃってるかもしれませんね。えと，これまで作ったプログラムの中に，次のようなちょっぴりおかしなIF文が，よくありました。たぶん，これから作るプログラムにも，たくさん現れるはず…です。

```
1720  IF  I=2  VX=-1
```

ねッ？　なんだかおかしい…でしょ。どこが，おかしいのかナ。よーく見ると，THENが抜けてるの。本当は，**IF　1=2　THEN　VX=-1**　じゃなくちゃいけませんよネ。けど，これは**IF　1=2　VX=-1**やっぱり，THENが抜けてます。

わあ，バグだ，バグだ！

そーなんです。間違いなんです本当は…。でも，ファミリーベーシックではERRORが出ません。まさか，こんなヘンな使い方するなんて思いもしなかったんでしょうねー。

コレを利用して，メモリ不足のときはIF文のTHENをはぶいちゃう。すると，2バイト節約できます。節約，節約…。

バイト数	サンプル・プログラム
19	1　IF　A=0　THEN　B=1
17	1　IF　A=0　B=1
15	1　IFA=0B=1

メモリの節約方法を説明しています。最高4バイト節約できます。

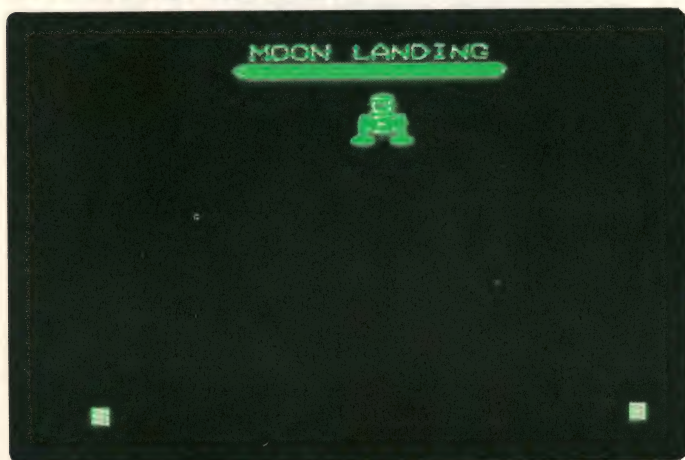
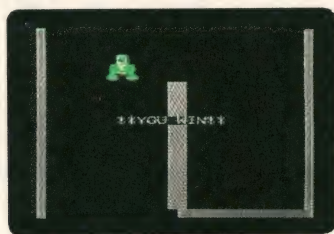
BGグラフィック画面06

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
0	K62						*	*		ク	ル	ク	ル	2	5	／	パ	ズ	ル		*	*						K62
1	K62						K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52						K62
2	K62																											K62
3	K62																											K62
4	K62				I60			I70	I60			I70	I60			I70	I60			I70	I60			I70				K62
5	K62				J20	0	1	J20	J20	0	2	J20	J20	0	3	J20	J20	0	4	J20	J20	0	5	J20				K62
6	K62				J00	J30	J30	J10	J00	J30	J30	J10	J00	J30	J30	J10	J00	J30	J30	J10	J00	J30	J30	J10				K62
7	K62				I60			I70	I60			I70	I60			I70	I60			I70	I60			I70				K62
8	K62				J20	0	6	J20	J20	0	7	J20	J20	0	8	J20	J20	0	9	J20	J20	1	0	J20				K62
9	K62				J00	J30	J30	J10	J00	J30	J30	J10	J00	J30	J30	J10	J00	J30	J30	J10	J00	J30	J30	J10				K62
10	K62				I60			I70	I60			I70	I60			I70	I60			I70	I60			I70				K62
11	K62				J20	1	1	J20	J20	1	2	J20	J20	1	3	J20	J20	1	4	J20	J20	1	5	J20				K62
12	K62				J00	J30	J30	J10	J00	J30	J30	J10	J00	J30	J30	J10	J00	J30	J30	J10	J00	J30	J30	J10				K62
13	K62				I60			I70	I60			I70	I60			I70	I60			I70	I60			I70				K62
14	K62				J20	1	6	J20	J20	1	7	J20	J20	1	8	J20	J20	1	9	J20	J20	2	0	J20				K62
15	K62				J00	J30	J30	J10	J00	J30	J30	J10	J00	J30	J30	J10	J00	J30	J30	J10	J00	J30	J30	J10				K62
16	K62				I60			I70	I60			I70	I60			I70	I60			I70	I60			I70				K62
17	K62				J20	2	1	J20	J20	2	2	J20	J20	2	3	J20	J20	2	4	J20	J20	2	5	J20				K62
18	K62				J00	J30	J30	J10	J00	J30	J30	J10	J00	J30	J30	J10	J00	J30	J30	J10	J00	J30	J30	J10				K62
19	K62																											K62
20	K62																											K62

スーパー月面着陸ゲーム

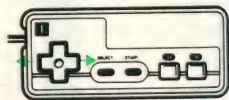
7 月面着陸

ゲームの遊び方

月面着陸船のエンジンが，故障しました。逆噴射は**不可能**…。残る方法は，ただひとつ。**緊急基地**に着陸するだけです。**ところが**，基地はふかーい谷底にあって，場所もハッキリと分らないんです。無事，帰還できるか！

テレビ画面の上の方にあるのは，月面着陸船です。谷にぶつからないように上手に着陸してネ

無事に着陸できたー！ ワーイワーイ!!

まず，BGグラフィックを入れて，それから，BASICプログラムを入力してネ。

このゲームは，月面着陸船をふかいふかーい谷底まで，ちゃんと着陸させるんです。谷はとっても複雑で，ゴチャゴチャいりくんでるの。だから，なかなか**上手に着陸できません**。

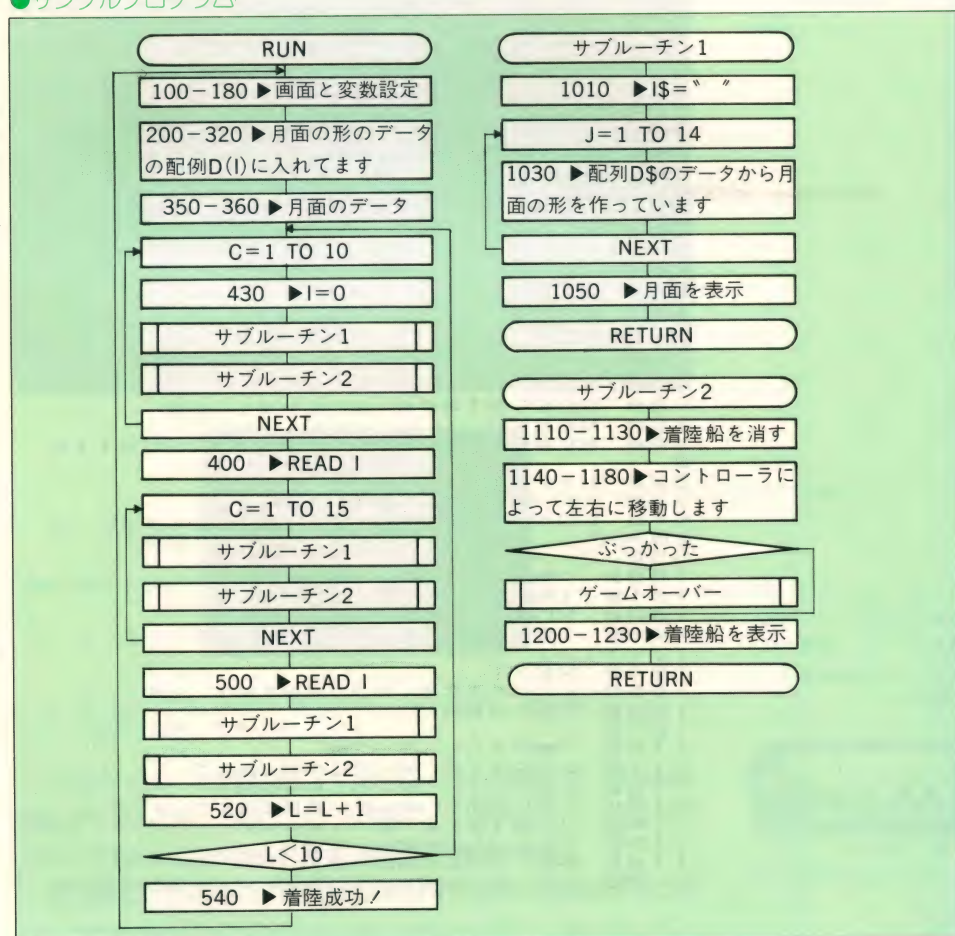
着陸船は，コントローラで左右に動かすことができます。谷にぶつかっちゃうとゲームオーバーになっちゃいますー。がんばって！

コントローラを左右に動かすと，着陸船は動きます

200行から320行は，月面の形のデータです。350行と360行は月面のデータを出す順番です。

サブルーチン１では，指定された月面データを画面に表示してます。変数Ｉに，表示したい形の番号を入れて呼びます。

```
100  '●● GAME/07(チアキ+カナエ) ●●
110  DIM D$(12),A$(3)
130  PALETB 0,13,16,33,2
170  A$(0)="  ":A$(1)=CHR$(25
3)+CHR$(253):A$(2)=CHR$(253)
+" ":A$(3)=" "+CHR$(253)
180  X=13:RESTORE
200  D$(0)="20000000000003"
210  D$(1)="20000032000003"
220  D$(2)="11111110000003"
230  D$(3)="20000001111111"
240  D$(4)="20001000010003"
250  D$(5)="20001111111111"
260  D$(6)="11111000011111"
270  D$(7)="11111111110003"
280  D$(8)="20010032001003"
290  D$(9)="20011111111111"
300  D$(10)="11110011111111"
310  D$(11)="11111111001111"
320  D$(12)="11111111111003"
330  '
350  DATA 1,3,1,3,1,2,1,3,1,2
,4,7,4,5,4,5,4,7,4,6,4,5,4,6
,4,6,8,12,8,12,8,11
360  DATA 8,9,8,12,8,10,8,10,
 8,11,8,9
370  '
400  '●メインルーチン●
410  L=0
420  FOR C=1TO10
430  I=0:GOSUB1000:GOSUB1100
440  NEXT
460  READI
470  FOR C=1TO15
480  GOSUB1000:GOSUB1100
490  NEXT
500  READI:GOSUB1000:GOSUB110
0
520  L=L+1
530  IF L<20 GOTO420
540  LOCATE8,10:PRINT"**YOU W
IN**";:INPUT"",A:GOTO120
550  '
1000  '●サブルーチン1●          データを月面に変換
1010  I$=""
1020  FOR J=1TO14
1030  I$=I$+A$(VAL(MID$(D$(I)←
,J,1)))
1040  NEXT
1050  LOCATE0,22:PRINT I$←
1060  RETURN                     月面を表示してます
1070  '
1100  '●サブルーチン2●
1110  LOCATE X,2:PRINT"    ";
1120  LOCATE X,3:PRINT"    ";
1130  LOCATE X,4:PRINT"    ";
1140  J=STICK(0):VX=0
1150  IF J=2VX=-2 ┐
1160  IF J=1VX= 2 ┘──コントロール操作
1170  IF X+VX=-1ORX+VX=25VX=0
```

```
1180  X=X+VX
1190  IF  SCR$(X,5)<>"  "  OR  SC
R$(X+2,5)<>"  "  GOTO1400
1200  LOCATE  X+1,3:PRINT"H1";
1210  '3:PRINTCHR$(209);
1220  'PRINTCHR$(236);CHR$(193);CHR$(235);
1230  'PRINTCHR$(233);"  ";CHR$(233);
1240  RETURN
1250  '
1400  '●ゲームオバー●
1410  LOCATE7,10:PRINT"**GAME
 OVER**";
1420  INPUT"",A:GOTO120
```

●サンプルプログラム

プログラムの説明

キャラクタだって，
主人公になれるんです

なんだか，思うように作れませんでした…。ちーちゃんのアイデアは，もっと面白そうだったのに。ごめんなさい…。

ファミリーベーシックでは，ふつうスプライト（アニメキャラクタ）が主人公ですよネ。動き回ったり，飛んだり跳ねたり…。

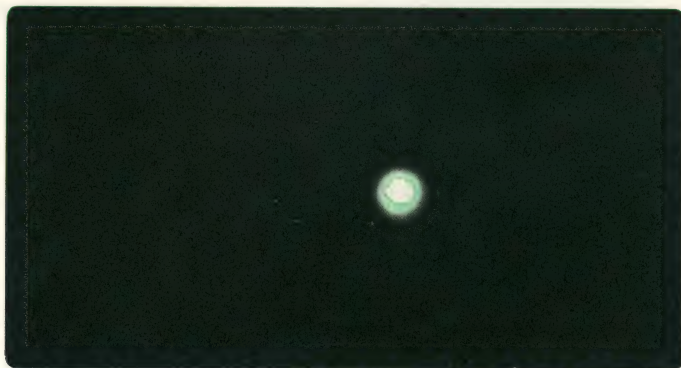
でも。

このプログラムでは，ごくごく普通のキャラクタ（文字）が，主人公の「月面着陸船」の役をしてるんです。

ところで今回は，普通のキャラクタの動かし方を説明したいナ，て思います。

スプライトとキャラクタは，動かし方がちょっぴり違います。次のプログラムは，スプライトを動かすサンプル…です。

コントローラを操作してくださいね。ニタニタが動き回ります。でも，画面の外に出ちゃうとエラーになります。

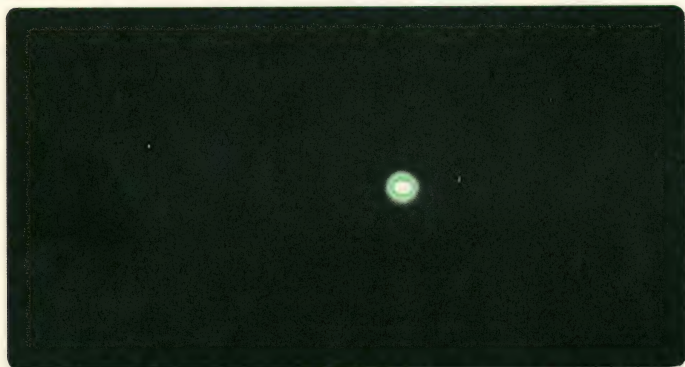
```
100  CGSET1.2:CLS:SPRITE ON
110  X=13:Y=10
120  DEF SPRITE 0,(1,1,0,0,0)
="XYZ「"  ←スプライト0を「ニタニタ」にします
200  I=STICK(0)
210  IF I=2 THEN X=X-1
220  IF I=1 THEN X=X+1
230  IF I=8 THEN Y=Y-1
240  IF I=4 THEN Y=Y+1
250  SPRITE 0,X*8+16,Y*8+22  ←スプライト0を表示
260  GOTO 200
```

スプライトの場合，表示したい位置を次々に変えてけば，ちゃあ

んと動いてくれます。

でもね，キャラクタの場合は，そう簡単にはゆきません。うまく動いてくれないんです。たとえば，次のプログラムみたく…。

```
100 CGSET1.2:CLS:SPRITE ON
110 X=13:Y=10
200 I=STICK(0)
210 IF I=2 THEN X=X-1
220 IF I=1 THEN X=X+1
230 IF I=8 THEN Y=Y-1
240 IF I=4 THEN Y=Y+1
250 LOCATE X,Y:PRINT CHR$(20
7);
260 GOTO 200
```

←●を表示します

コントローラで〈●〉を動かしてください。でも，動いた跡が残っちゃうけど…。

ね？　キャラクタだと動いた跡，残っちゃう。これじゃあ　困っちゃいますよネ。

だから，キャラクタを動かす時は，表示する前に動いた跡を消してから，表示すると良いみたい…です。ただし，ファミリーベーシックって，キャラクタを表示するの少し遅いから使い方には注意しましょう…。

```
100 CGSET1.2:CLS:SPRITE ON
110 X=13:Y=10
150 LOCATE X,Y:PRINT" ";
200 I=STICK(0)
210 IF I=2 THEN X=X-1
220 IF I=1 THEN X=X+1
230 IF I=8 THEN Y=Y-1
240 IF I=4 THEN Y=Y+1
250 LOCATE X,Y:PRINT CHR$(20
7);
260 GOTO 150
```

コントローラで〈●〉を動かしてください。今度は，ちゃあーんと動いた跡，消えます。

BGグラフィック画面07

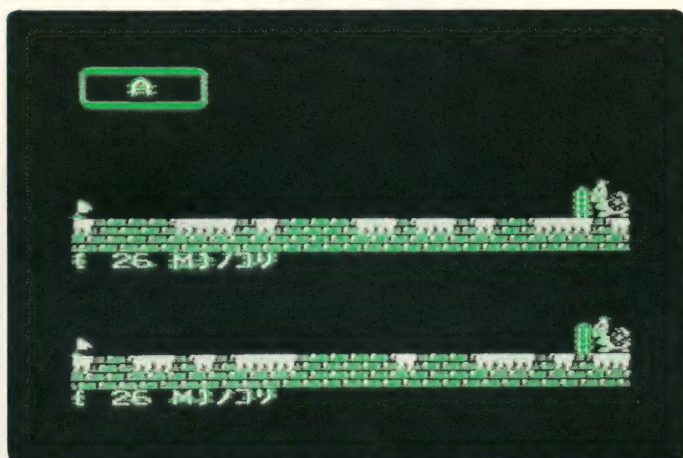
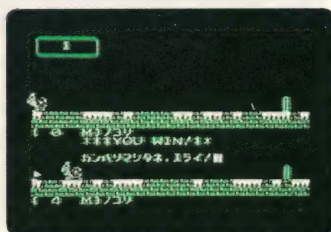
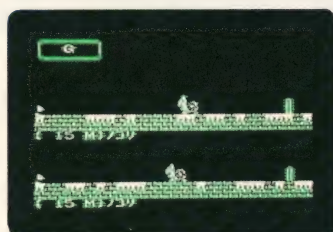
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
0								J60	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J70							
1									M	O	O	N		L	A	N	D	I	N	G								
2								J60	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J70							
3																												
4																												
5																												
6																												
7																												
8																												
9																												
10																												
11																												
12																												
13																												
14																												
15																												
16																												
17																												
18																												
19																												
20																												

コンピュータのカメさんと競走です

8 よい子のかけっこ競走

ゲームの遊び方

よーい，ドン!!

で，コンピュータの『カメさん』と**カケッコ**してネ。コンピュータのカメさんは，１歩１歩キチンと進みます。もう，カメさんの**カガミ**なんです。このカメさんに，追いつけ追いこせ！

テレビ画面の上のカメさんが，あなたのカメさんです。下のカメさんは，コンピュータのカメさんです。
画面の左上に，アルファベットがでてくるんですよー。

ガンバレッガンバレッ。

わーい，コンピュータのカメさんに勝っちゃった！

ゲームを遊ぶ前に，BGグラフィックを入れて，BASICプログラムを**入力**しておいてネ。

このゲームは，コンピュータのカメさんとかけっこするんです。ね，面白そうでしょー。みなさんのカメさんをかけっこさせるにはテレビ画面左上にでてくる文字とおんなじ文字，キーボードから入力すればいいの。タイプミスが無いほどスピードが速くなってゆきます。でも，**なかなか勝てないんですよ。**

画面の左上に出てきたアルファベットと同じ文字を，キーボードから入れます。

130行から160行ではカメさんをスタート地点に表示してます。170行では「ピッピッピッポッ」の音を出してます。190行から220行では MOVE 文の設定をしてます。

入力したキャラクタが合ってるかどうかを調べてます。

サブルーチン2では，あなたの動かすカメさんを走らせています。

```
100  '●● GAME/08(チアキ+カナエ) ●●
110  CGSET1,2:VIEW:SPRITE ON
120  ERA 0,1
130  DEF SPRITE 0,(3,1,0,0,0)
=CHR$(184)+CHR$(185)+CHR$(18
6)+CHR$(187)
140  DEF SPRITE 1,(3,1,0,0,0)
=CHR$(184)+CHR$(185)+CHR$(18
6)+CHR$(187)
150  SPRITE 1,224,78
160  SPRITE 0,224,142
170  PLAY"C0R7C0R7C0R7A2"
180  SPRITE 0:SPRITE 1
190  DEF MOVE(0)=SPRITE(13,7,
1,4,0,3)
200  POSITION 0,224,142
210  DEF MOVE(1)=SPRITE(13,7,
1,1,0,3)
220  POSITION 1,224,78
230  MOVE 1:X=224:V=10
240  '
300  '●メインルーチン●
310  O=-1
320  FOR I=26 TO 1 STEP-1
330  MOVE 0
350  I$=CHR$(RND(26)+65):P=0
360  LOCATE 3,1:PRINT I$;
370  FOR T=1 TO 300
380   J$=INKEY$
390   IF J$<>"" GOSUB1000
410  NEXT
420  IF P=2 GOSUB1100
440  LOCATE 1,19:PRINTI;"  ";
450  LOCATE 1,11:PRINT (X-16)
/8;"  ";
470  IF X>224 OR X<20 GOTO130
0
480  NEXT
490  GOTO1200
500  '
1000  '●サブルーチン1●
1010  IF P>0 THEN RETURN
1020  IF I$=J$ THEN V=V+2:P=2
1030  IF I$<>J$ THEN V=14:P=1
1040  RETURN
1050  '
1100  '●サブルーチン2●
1110  X=XPOS(1)
1120  DEF MOVE(1)=SPRITE(13,7
,1,V/5,0,3)
1130  POSITION 1,X,78
1140  MOVE 1
1150  RETURN
1160  '
1200  '●ゲームオーバー1●
1210  LOCATE 5,12:PRINT"**GAM
E OVER**";
1220  LOCATE 5,14:PRINT"ザンネン
デシタ.オシイ!!";
1230  INPUT"",A:GOTO110
1240  '
```

（370行の 300：この値を大きくするとキー入力時間がのびます）

（1030行の 14：この値を大きくするとあなたのカメの動く量がふえます）

```
1300 '●ケ゛ームオーハ゛ー2●
1310 LOCATE 5,12:PRINT"***YO
U WIN***";
1320 LOCATE 5,14:PRINT"カ゛ンハ゛
リマシタネ.エライ!";
1330 INPUT"",A:GOTO110
```

●サンプルプログラム

RUN
100 ▶REM文
120−180▶SPRITE文の設定をしてます
190−220▶MOVE文の設定をしてます
230 ▶変数の設定
I=26 TO 1 STEP−1
330▶コンピュータのカニさんを動かす
T=1 TO 300
INKEY$<>" "
サブルーチン1
NEXT
P=2
サブルーチン2
440−450▶スコア表示
ゴールイン?
ゲームオーバー2
NEXT
ゲームオーバー2
サブルーチン1
P>0
RETURN
1020 ▶正しくキー入力されたら P=2
1030 ▶間違ってキー入力されたら P=1
RETURN
サブルーチン2
1110−1140▶あなたのカニさんを動かす
RETURN
ゲームオーバー1
1212−1230▶ゲームオーバーのメのメッセージを表示します
RETURN
ゲームオーバー2
1310−1330▶YOU WIN のメッセージを表示します
RETURN

プログラムの説明

SPRITEと
MOVE文の使い方

このプログラムでは，特に変わったこと，してません。もう，とても素直に作っただけ…なんです。プログラムを見てもらえば分かる，と思います。

なので，このプログラムの説明！，ていうのは無いんですネ。困ってしまいます…。

そこで，**SPRITE文とMOVE文の使い方**を説明してみたいと思います。まずは，SPRITE文から…。

SPRITE文は，**スプライト（アニメキャラクタ）をテレビ画面に出すための命令**，です。**DEF SPRITE，SPRITE ON／OFF**の5種類の命令を，上手に使います。

命令	意味
DEF SPRITE	スプライトの形を決める命令
SPRITE	決めたスプライトを表示する命令
SPRITE ON	スプライトの表示を可能にします
SPRITE OFF	スプライトの表示を止めます

（くわしくはマニュアルを…）

それじゃあ，実際にSPRITE文を使って，画面にスプライトを表示してみますネ。

最初に，画面をスプライトを表示できるように，**SPRINE ON**しておきます。

```
10 SPRITE ON
```

スプライトを表示できるようにします。

つぎは，**DEF SPRITE文で，表示したいアニメキャラクタの形を決めます**。そうですねー，前を向いたペンギンさんにしたいナ，て思います。どうです？

●文字列
文字（キャラクタ）がいくつか並んだもの。

前向きのペンギンさんの形の文字列，CHR$（104）+CHR$（105）+CHR$（106）+CHR$（107）になるそうです…。

マニュアルの「キャラクタテーブルA」を見てくださいネ。あ，

CHR$を使う代わりに，〝ケコサシ〟，でも良いみたい…。
　というわけで，次のようになります。

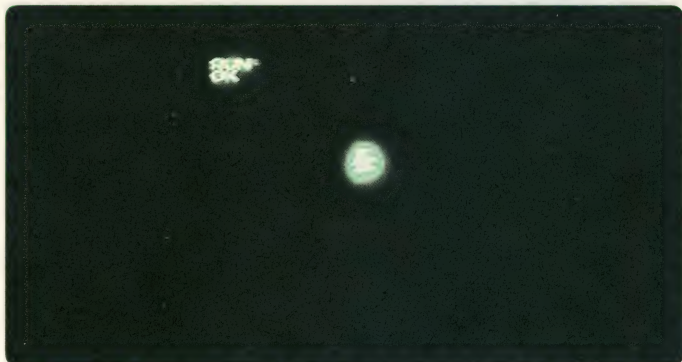
10でスプライトを表示可能にして，20行でペンギンさんに決めてます。

```
1Ø SPRITE ON
2Ø DEF SPRITE Ø,(Ø,1,Ø,Ø,Ø)=
CHR$(1Ø4)+CHR$(1Ø5)+CHR$(1Ø6
)+CHR$(1Ø7)
```

　そして，SPRITE文を使って…，画面にスプライトを表示します。表示する位置は，タテ=100，ヨコ=100でいいかナ。
　動かすと，画面のだいたい真ん中にペンギンさんが表示されます。

30行で（100，100）にスプライトを表示してます。

```
1Ø SPRITE ON
2Ø DEF SPRITE Ø,(Ø,1,Ø,Ø,Ø)=
CHR$(1Ø4)+CHR$(1Ø5)+CHR$(1Ø6
)+CHR$(1Ø7)
3Ø SPRITE Ø,1ØØ,1ØØ
```

画面の真ん中にペンギンさんが表示されました

　今度は，MOVE文。
MOVE文も，SPRITE文とおんなじスプライトを表示するための命令…なんです。でも，ちょっぴり違います。MOVE文は，表示するだけじゃなくて，動かすことできちゃいます。だから，MOVE（動かすという意味デス）文，なんですネ…。
DEF　MOVE，MOVE，POSITIONといった命令を，上手に使います。

命令	意味
DEF MOVE	スプライトの形と動きを決めます
MOVE	スプライトの動きを開始
CUT	スプライトの動きを停止
ERA	スプライトの動きを停止して消す
POSITION	動き始めの位置を決めます
XPOS	スプライトのヨコ位置を調べます
YPOS	スプライトのタテ位置を調べます
MOVE (n)	動きが終わったたかどうかを調べます

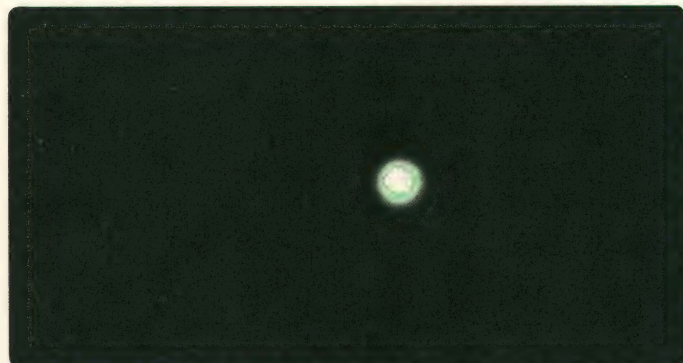
まず，DEF　MOVE文でスプライト（アニメキャラクタ）を選んで，動きも決めます。えと，カメさんが右へトコトコ動くようにしたいと思います…。

```
10 SPRITE ON
20 DEF MOVE(0)=SPRITE(13,3,1
,25,0,2)
```

10行でスプライトを表示可能にして，20行でカメさんの動きを設定してます。

スプライトの動きと形を決めたら，今度はPOSITION文を使って，位置を決めます。画面の真ん中へんの（100，100）にしましょうね。RUNすると，画面の真ん中から右へ，カメさんがトコトコ歩きます…。

```
30 POSITION 0,100,100
40 MOVE 0
```

30行で動き始めの場所を決めて，40行で動かします。

BGグラフィック画面08

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
0	K72	K52	K52	K52	K52	K52	K52																					
1	K62			A			K62																					
2	L12	K52	K52	K52	K52	L02	L22																					
3																												
4																												
5																												
6																												
7																										J40		
8	F70																									J20		
9	F23	F33	F33	F33	F33	F23	F23	F23	F33	F23	F33	F33	F33	F33	F23	F23	F33	F23	F23	F33	F23	F33	F33	F23	F23	F23	F33	F23
10	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33
11	[		2	6		M	]	／	ユ	リ																		
12																												
13																												
14																												
15																										J40		
16	F70																									J20		
17	F23	F33	F33	F23	F23	F33`	F23	F33	F23	F23	F23	F33	F33	F33	F33	F33	F23	F33	F33	F33	F23	F23	F23	F33	F23	F33	F23	F23
18	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33
19	[		2	6		M	]	／	ユ	リ																		
20																												

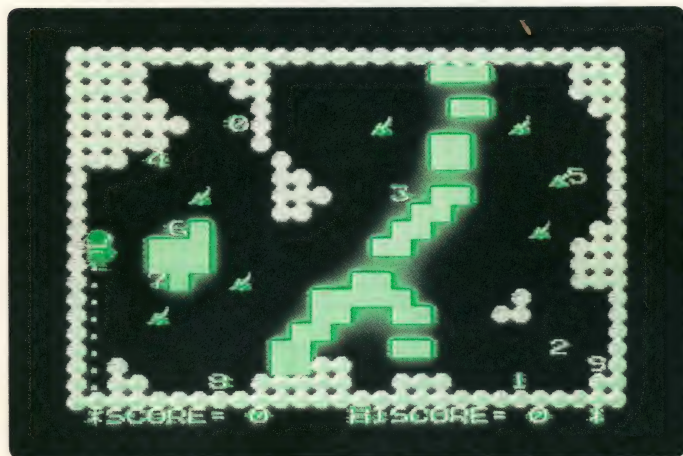
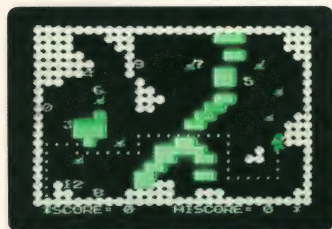
公園にちらばった数字を集めるゲームです

9 散歩でよいしょ!

ゲームの遊び方

公園にちらばった数字を，順番にひろってネー。

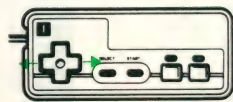
公園には，**川**とか**森**，それに**岩**なんかがゴロゴロ…。ぶつかっちゃうと，だめなの。

なかなか，むつかしいけど，ガンバッてみてねっ!

画面の左下にいるのは，みなさんが動かす『マリオ』です。画面には，0から9までの数字がありますよね。

画面の障害物にぶつかっちゃったり，足跡にぶつかっちゃうとゲームオーバーなんですー。気を付けてネ。

0から9までの数字を取れば，いいの。

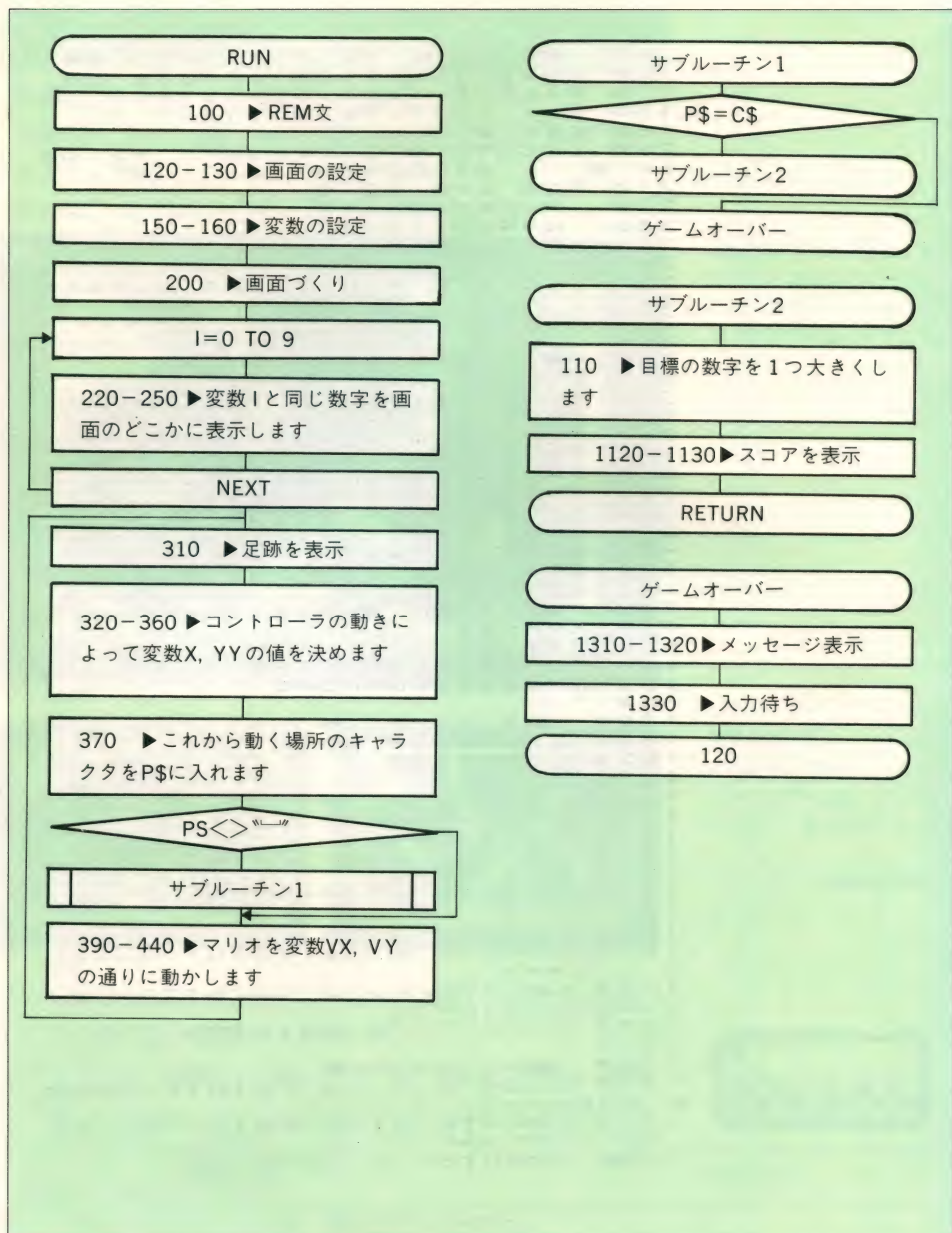
ゲームを遊ぶ前に，BGグラフィックを入れて，それからBASICプログラムを入力しておいてくださいネ。

このゲームは，公園の中を散歩して，0から9までの数字を集めるんですー。うーん，なんで公園で数字を集めなきゃいけないの?とにかく，0から9まで集めてくださいネ。『マリオ』は，コントローラで上下左右に動かせます。

んっ　ちょっぴりむつかしめ，だけどガンバってね!

コントローラで，マリオを動かします。

```
100  '●● GAME/09(チアキ+キムラ) ●●
120  CGSET1,2:VIEW:SPRITE ON
130  PALETB 1,13,48,55,41
150  X=1:Y=19:VX=0:VY=-1
160  C$="0"
200  LOCATE 0,21:PRINT" *SCOR
E= 0     HISCORE=";H;"   *";
210  FOR I=0 TO 9
220   M=RND(26)+1
230   N=RND(19)+1
240   IF SCR$(M,N)<>" " GOTO2
20
250   LOCATE M,N:PRINT CHR$(4
8+I);
260  NEXT
270  '
300  '●メインルーチン●
310  LOCATE X,Y:PRINT".";
320  I=STICK(0)
330  IF I=2 THEN VX=-1:VY=0
340  IF I=1 THEN VX= 1:VY=0
350  IF I=8 THEN VY=-1:VX=0
360  IF I=4 THEN VY= 1:VX=0
370  P$=SCR$(X+VX,Y+VY)
380  IF P$<>" " GOSUB1000
390  M=VX+VY+2-((VX+VX)<VY)*(
-VX-VX-VY-VY+4)
400  DEF MOVE(0)=SPRITE(0,M,1
,4,0,2)
410  POSITION 0,X*8+12, Y*8+20
420  X=X+VX
430  Y=Y+VY
440  MOVE0:PLAY "T1#C0"
450  PAUSE9:GOTO300
460  '
1000  '●サブルーチン1●
1010  IF P$=C$ GOTO1100
1020  GOTO1300
1030  '
1100  '●サブルーチン2●
1110  C$=CHR$(ASC(C$)+1)
1120  S=S+100
1130  LOCATE 8,21:PRINT S;
1140  PLAY"G1A1"
1150  IF C$<":" RETURN
1160  '
1200  '●サブルーチン3●
1210  LOCATE 6,10:PRINT"***YO
U WIN***";
1220  LOCATE 6,12:PRINT"ワァ,サイ
ゴマデデキマシタ";
1230  INPUT"",A:GOTO120
1240  '
1300  '●ゲームオーバー●
1310  LOCATE 6,10:PRINT"**GAM
E OVER**";
1320  LOCATE 6,12:PRINT"ザンネン
.シッパイデスネ";
1330  INPUT"",A:GOTO120
```

●サンプルプログラム

RUN
100 ▶REM文
120−130 ▶画面の設定
150−160 ▶変数の設定
200 ▶画面づくり
I=0 TO 9
220−250 ▶変数Iと同じ数字を画面のどこかに表示します
NEXT
310 ▶足跡を表示
320−360 ▶コントローラの動きによって変数X, YYの値を決めます
370 ▶これから動く場所のキャラクタをP$に入れます
PS<>"␣"
サブルーチン1
390−440 ▶マリオを変数VX, VYの通りに動かします
サブルーチン1
P$=C$
サブルーチン2
ゲームオーバー
サブルーチン2
110 ▶目標の数字を1つ大きくします
1120−1130▶スコアを表示
RETURN
ゲームオーバー
1310−1320▶メッセージ表示
1330 ▶入力待ち
120

プログラムの説明

スピードを速くするには，

こんな風にします

このプログラムも，ふつーに作りました。なので，やっぱり特に変わったこと，してないんです…。うーん，説明することなくて，困ります。

しかたないから，ゲームのスピードを上げる方法，について説明してみたいと思います。でも，わたしもそんなに詳しくないから，そのつもりで読んでくださいネ…。

①掛け算・割り算は，なるべく使わないように

掛け算と割り算は，足し算・引き算と比べてスピードが遅いんです。ちょっと信じられないかも…。でも，本当です。

なので，×2とか×3は足し算に代えましょう。少し，スピードが速くなります。

このプログラムは，実行し終わるまでに7.11秒かかります。

```
10  FOR I=1 TO 1000
20  J=I*2:J=I*2:J=I*2:J=I*2
30  NEXT
40  END
```

このプログラムは，実行し終わるまでに5.64秒かかります。

```
10  FOR I=1 TO 1000
20  J=I+I:J=I+I:J=I+I:J=I+I
30  NEXT
40  END
```

②複雑な計算は，初めに計算しておきます

複雑な計算は，なるべくその場で計算しないで，その前に用意しておいた方が良いみたい…。初めに計算しておいて，配列に入れておくんです。でも，この方法は，どんな場合でも使える，ていうわけじゃないから，気を付けてくださいね。すごーく複雑な計算の時，役に立ちます。

このプログラムは，実行し終わるまでに15.79秒かかります。

```
100 FOR C=1 TO 100
110 FOR I=1 TO 10
120 J= (I*I+3)*2+(I MOD 2=0)
130 J= (I*I+3)*2+(I MOD 2=0)
140 J= (I*I+3)*2+(I MOD 2=0)
150 NEXT
160 NEXT
170 END
```

```
10  DIM  D(10)
20  FOR  I=1  TO  10
30  D(I)=(I*I+3)*2+(I  MOD  2=0
)
40  NEXT
100  FOR  C=1  TO  100
110  FOR  I=1  TO  10
120  J=D(I)
130  J=D(I)
140  J=D(I)
150  NEXT
160  NEXT
170  END
```

このプログラムは，実行し終わるまでに12.16秒かかります。

③プログラムを圧縮します

マルチステートメントを使って，1行にたくさん命令を入れるとスピードが速くなります。でも，ほどほどにしないとゴチャゴチャになっちゃいますヨ。気を付けてね…。

```
10  FOR  I=1  TO  3000
20  J=I+1
30  J=I+2
40  J=I+3
50  NEXT
60  END
```

このプログラムは，実行し終わるのに5.96秒かかります。

```
10  FORI=1TO3000:J=I+1:J=I+2:
J=I+3:NEXT:END
```

このプログラムは，実行し終わるのに4.76秒かかります。

④変数名は，なるべく1文字にします

変数名があんまり長いと，スピードが遅くなります。やっぱり，読むのに時間がかかっちゃうんだと思います。なるべく，1文字の変数名を使いましょうね。

あッ，でも，これはスピードを速くしたい場合です。かならずしなきゃいけない，わけじゃありません…。

```
10  FOR  COUNT=1  TO  1000
20  ALIEN=COUNT+COUNT
30  BOOKS=COUNT+COUNT
40  DINER=COUNT+COUNT
50  NEXT
60  END
```

このプログラムは，実行し終わるまでに12.39秒かかります。

```
10  FOR  C=1  TO  1000
20  A=C+C
30  B=C+C
40  D=C+C
50  NEXT
60  END
```

このプログラムは，実行し終わるまでに11.34秒かかります

⑤ＩＦ文では，ANDを使わないように

IF文の条件は，なるべくANDを使わない方が良いそうです。特に，IF X=255 AND P=ASC("●") AND VX=−1 AND I=1 THEN，なんていうのはやめた方がいいと思います。

１つのIF文にたくさん条件を詰め込まないで…。2つとか3つとかに分けるようにしましょう。

このプログラムは，実行し終わるのに5.70秒かかります。

```
10  FOR I=1 TO 1000
20  J=STICK(0)
30  IF J=8 AND X=0 THEN X=1
40  IF J=4 AND X=1 THEN X=0
50  NEXT
60  END
```

このプログラムは，実行し終わるのに4.07秒かかります。

```
10  FOR I=1 TO 1000
20  J=STICK(0)
30  IF J=8 GOSUB 100
40  IF J=4 GOSUB 100
50  NEXT
60  END
70  '
100  IF X=0 THEN X=1
110  IF X=1 THEN X=0
120  RETURN
```

BGグラフィック画面09

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
0	G71	G71	G71	G71	G71	G71	G71	G71	G71	G71	G71	G71	G71	G71	G71	G71	G71	G71	G71	G71	G71	G71	G71	G71	G71	G71	G71	G71
1	G71	G71	G71	G71				G71	G71	G71									M60	M60	M60					G71	G71	G71
2	G71	G71	G71	G71					G71	G71																G71	G71	G71
3	G71	G71	G71	G71	G71					G71										M60	M60						G71	G71
4	G71	G71	G71	G71	G71	G71				G71						G02							G02					G71
5	G71	G71	G71	G71	G71					G71									M60	M60								G71
6	G71	G71	G71								G71								M60	M60								G71
7	G71										G71	G71													G02			G71
8	G71						G02				G71	G71	G71						M60	M60								G71
9	G71										G71	G71						M60	M60									G71
10	G71						M60										M60	M60						G02			G71	G71
11	G71				M60	M60	M60									M60	M60									G71	G71	G71
12	G71				M60	M60	M60																			G71	G71	G71
13	G71					M60			G02						M60	M60										G71	G71	G71
14	G71												M60	M60	M60	M60	M60						G71					G71
15	G71				G02								M60	M60			M60	M60				G71	G71					G71
16	G71											M60	M60															G71
17	G71										M60	M60					M60	M60										G71
18	G71		G71								M60	M60	G71	G71														G71
19	G71		G71	G71						G71	G71	G71	G71	G71			G71	G71	G71								G71	7
20	G71	G71	G71	G71	G71	G71	G71	G71	G71	G71	G71	G71	G71	G71	G71	G71	G71	G71	G71	G71	G71	G71	G71	G71	G71	G71	G71	7

コンピュータとじゃんけんしましょ！

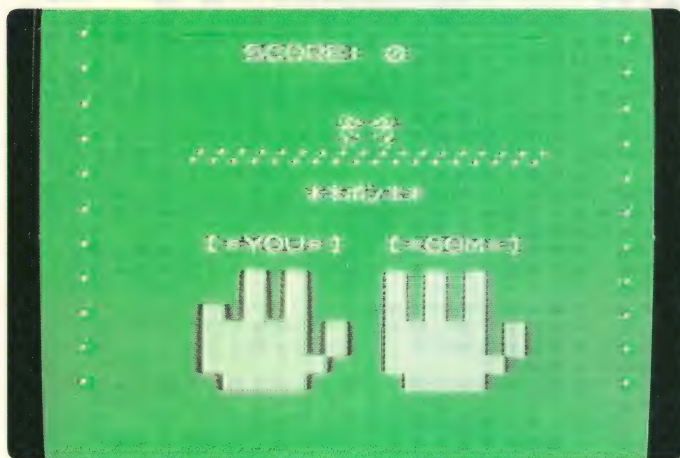
10 じゃんけん・ポン

ゲームの遊び方

ジャンケン…，ポン！

『ファミコン』と，じゃんけんしてネ。みなさんも，**ファミコンとじゃんけん**したことないでしょー。

でも，アトだしとかチョンボは，しないでね。

コンピュータとジャンケンをしてるところ，ですー。左があなたの手，右がコンピュータの手です。

スタートしたばかりの，テレビ画面です。まだ，あなたとコンピュータの手が出てないですネ。

ゲームオーバー

ゲームを遊ぶ前に，BGグラフィックを入れて…。それから，BASICプログラムを入力してくださいネ。

このゲームは，コンピュータとジャンケンをするんでーす。

スタートすると，「ジャンケン・ポン」て画面に出てくるから，出すのを決めて…。グーなら左，チョキなら下，パーなら右です。で，勝ったらAボタン，負けちゃったらBボタンを，スグに押してネ。ちゃあんと押さないと，ゲームオーバーですっ!!

```
100  '●● GAME/10(チアキ+カナエ) ●●
120  CGSET1,0:VIEW:SPRITE ON
150  S=0:SY=0:SC=0:T=100
160  PALETS 0, 5,54,37,2
200  DEF SPRITE 0,(0,1,0,1,0)
=CHR$(5)+CHR$(4)+CHR$(7)+CHR
$(6)
210  DEF SPRITE 1,(0,1,0,0,0)
=CHR$(4)+CHR$(5)+CHR$(6)+CHR
$(7)
220  DEF SPRITE 2,(3,1,0,0,0)
=CHR$(192)+CHR$(193)+CHR$(19
4)+CHR$(195)
230  DEF SPRITE 3,(3,1,0,0,0)
=CHR$(192)+CHR$(193)+CHR$(19
4)+CHR$(195)
240  DEF SPRITE 4,(1,1,0,1,0)
=CHR$(185)+CHR$(184)+CHR$(18
7)+CHR$(186)
250  DEF SPRITE 5,(1,1,0,0,0)
=CHR$(184)+CHR$(185)+CHR$(18
6)+CHR$(187)
260  SPRITE 0,12*8+16,5*8+22
270  SPRITE 1,14*8+16,5*8+22
290  '
300  '●メインルーチン●
310  PALETB 0,5,48,5,2
320  PALETB 1,5,48,5,24
330  D$=""
350  LOCATE 11,9:PRINT"ジャンケン
!";
360  PLAY"D3R5D3R5C3"
370  LOCATE 11,9:PRINT"**ポン*
*";
380  D=0
390  FOR C=1 TO 50
400   I=STICK(0)
410   IF I>0 GOSUB1000
420  NEXT
430  IF D=0 THEN1500
450  I=D:GOSUB1100
460  LOCATE  6,14:PRINT I$;
470  LOCATE  6,15:PRINT J$;
480  I=RND(3)+1:GOSUB1100
490  LOCATE 15,15:PRINT I$;
500  LOCATE 15,15:PRINT J$;
510  PALETB 0, 5,48,49,2
520  PALETB 1,13,48,53,24
530  PAUSE T
540  D=0:PLAY"#A0"
550  FOR C=1 TO 50
560   I=STRIG(0)
570   IF I>0 GOSUB1200
580  NEXT
600  IF D=0 THEN1600
610  I=D:GOSUB1300
620  LOCATE 14,1:PRINT S;
630  PAUSE100
640  GOTO300
650  '
```

200行から250行では，スプライトの設定をしてます。260行と270行では，マリオを表示してます。

310行と320行では，手を消してます。

390行から520行ではジャンケンの表示をしてます。390行から420行で，コントローラを調べています。460行から520行は，手の表示です

サブルーチン1は，コントローラを調べるルーチンです。

サブルーチン2は，ジャンケンの手の形を決めるルーチンです。

サブルーチン3は，カチ・マケ・アイコの入力するルーチンです。

サブルーチン4は，カチ・マケ・アイコの表示文字を決めるルーチンです。もちろん表示もしています。

```
1000 '●ｻﾌﾞﾙｰﾁﾝ1●
1010 IF I=2 THEN D=1:D$="1"
1020 IF I=4 THEN D=2:D$="2"
1030 IF I=1 THEN D=3:D$="3"
1040 RETURN
1050 '
1100 '●ｻﾌﾞﾙｰﾁﾝ2●
1110 IF I=1 THEN I$="        ":
J$="        ":D$=D$+"1"
1120 IF I=2 THEN I$="  ■ ■":
J$="  ■ ■":D$=D$+"2"
1130 IF I=3 THEN I$="■ ■ ■":
J$="■ ■ ■":D$=D$+"3"
1140 RETURN
1150 '
1200 '●ｻﾌﾞﾙｰﾁﾝ3●
1210 IF I=8 THEN D=1
1220 IF I=4 THEN D=2
1230 IF I=12 THEN D=3
1240 RETURN
1250 '
1300 '●ｻﾌﾞﾙｰﾁﾝ4●
1310 IF I=1 THEN I$="* ｶﾁ *"
:S=S+10
1320 IF I=2 THEN I$="* ﾏｹ *"
1330 IF I=3 THEN I$="*ｱｲｺ!*"
1340 LOCATE 11,9:PRINT I$;
1350 D$=RIGHT$(D$,2)
1360 IF (D$="12" OR D$="23"
OR D$="31")AND I<>1 THEN1600
1370 IF (D$="13" OR D$="21"
OR D$="32")AND I<>2 THEN1600
1380 IF (D$="11" OR D$="22"
OR D$="33")AND I<>3 THEN1600
1390 T=T-3
1400 RETURN
1410 '
1500 '●ｹﾞｰﾑｵｰﾊﾞｰ1●
1510 LOCATE 7, 9:PRINT"** ｱﾄ
ﾀﾞｼﾃﾞｽ **";
1520 LOCATE 7,11:PRINT"ｽｺｼ ﾊ
ﾔｸｼﾏｼｮｳ";
1530 INPUT"",A:GOTO120
1540 '
1600 '●ｹﾞｰﾑｵｰﾊﾞｰ2●
1610 LOCATE 7, 9:PRINT"** ﾁｮ
ﾝﾎﾞﾃﾞｽ **";
1620 LOCATE 7,11:PRINT"ﾏﾁｶﾞｴ
ﾅｲﾖｳﾆｼﾃﾈ";
1630 INPUT"",A:GOTO120
```

●フローチャート

RUN
100－200 ▶REM文と画面の設定と変数の設定
200－250 ▶スプライト設定
260－270 ▶マリオを表示
310－320 ▶手を消します
330－370 ▶音を出しながらジャンケンポンします
C＝1 TO 5
STICK>0
サブルーチン1
NEXT
430 ▶D＝0ならゲームオーバーです
サブルーチン2
460－530 ▶手を表示
C＝1 TO 5
STICK>0
サブルーチン3
NEXT
600 ▶チョンボしたのならゲームオーバーです
サブルーチン4
620－630 ▶スコア表示
サブルーチン1
1010－1030▶コントローラの動きによって変数DかD$を設定
RETURN
サブルーチン2
1110－1130▶変数Iの値によって手の形を決めます
RETURN
サブルーチン3
1210－1230▶コントローラの動きによって変数Iを決めます
RETURN
サブルーチン4
1310－1330▶変数Iの値によってメッセージを決めます
1340 ▶メッセージ表示
1350－1390▶エラーチェック
RETURN
ゲームオーバー
1510－1520▶メッセージ表示
GOTO 120

プログラムの説明

プログラムを作りたいナ
て思ったら…

プログラムの説明は,もう,これで最後です。ここまでで終わり…なんです。

そこで,プログラムの上手な作り方,について説明したいな…て思います。でも,こーいうのってとても難しいから,心配です。上手に説明できなくっても,おこらないでネ…。

①どんなプログラムを作るかを,決めます

まず最初にやること…。

それは,どんなふーなプログラムを作るか,キチンと決めることなんだそうです。やっぱり,どんなプログラムを自分は作りたいか,ハッキリしてないと作りにくいんです。

②フローチャートを作ります

フローチャートなんていうと,なんだかムツカシそう…です。でもでも,本当はぜんぜんそんなことないんです。

フローチャートは,プログラムの流れ図です。プログラムの流れをなんとなく書いておけば,いいの…。それだけで,作るのがとてもラクになります。

だって,なーんともないのに,急にキーボードの前に座ってプログラム作っちゃう…なんて。なかなか,普通できません。

そこで,だいたいこんなふーに作ってゆくって,フローチャートを書いておくと,楽,なんですネ。

③うまく動かない時は,マイコンになっちゃう

プログラムを作ってて…。ぜんぜん,思った通りにプログラムが動いてくれないことがある,と思います。もう,こんな時って,イヤになっちゃいます…。

そんな時は,マイコンの気持ちになって考えてみると,とても良いみたい,です。1つ1つのプログラムの命令を,マイコンになったつもりで試してみるんです。そーすると,どこが間違ってたかが分ってくる,と思います。

④カンタンにして，考えてみましょう

マイコンの気持ちになっても，分からない時…。

こんな時は，プログラムを簡単にして考えてみると，いいみたい。

たとえば，エイリアンが10ピキ出てくるゲームだったら，１ピキ出てくるゲームのつもりで考えてみたり…。画面いっぱいの迷路だったら，ちっちゃい迷路のつもりで考えてみたり…。

つまり，大っきい物はちっちゃく，多い物は少なく，わけの分かんない物はカンタンに考えると，いいと思います。

⑤図を描いてみましょう

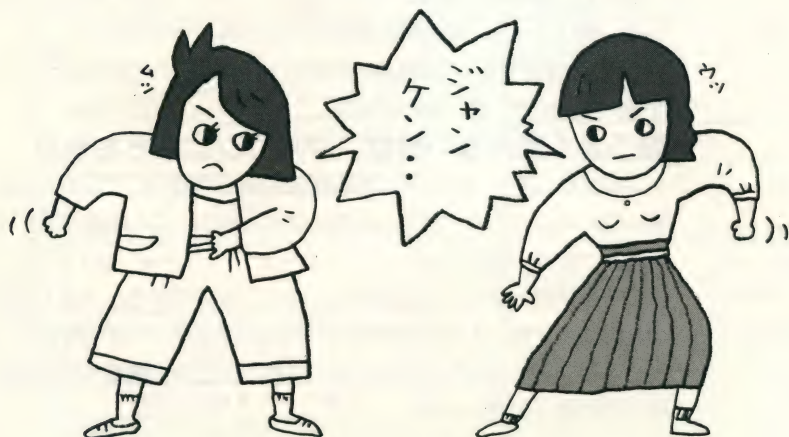
分からなくなったら，図を描く…。

とにかく，よく分かんなーいって思ったらスグに図を描いてみましょう。こーすると，ハッキリクッキリ，プログラムが見えてくるんです。

⑥イヤになったら、やめちゃう

もう，ぜんぜんできないよー，てイヤになったら，スグにやめて寝ちゃうの…。無理して作るよりも，このほうが効果的…です。楽しく作らなくちゃ，ソンです。

⑦簡単そうなプログラムから，作りましょう

無理をしないで，なるべく簡単なプログラムから作りましょう。

BGグラフィック画面10

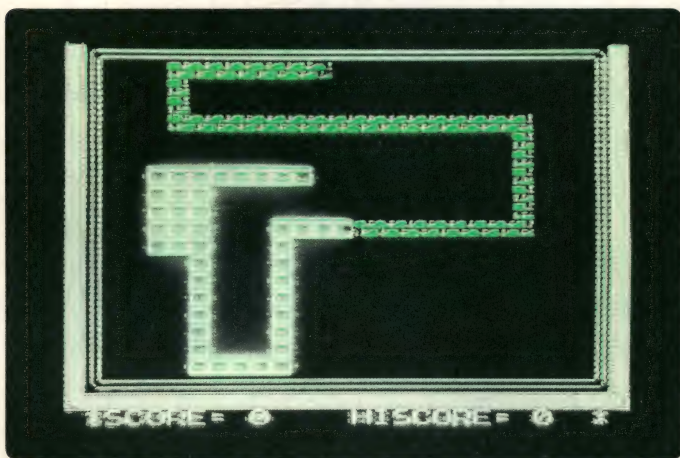
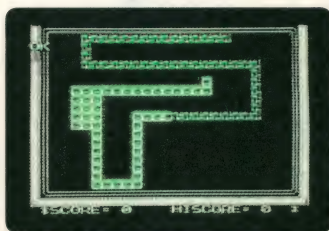
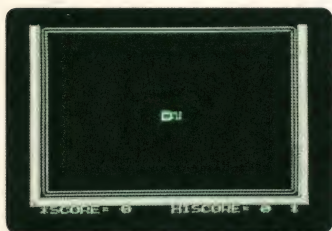
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
0	H72					K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50					H72
1									S	C	O	R	E	:		O												
2	H72																											H72
3																												
4	H72																											H72
5																												
6	H72																											H72
7						F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33					
8	H72																											H72
9																												
10	H72																											H72
11																												
12	H72						[	=	Y	O	U	=	]			[	=	C	O	M	=	]						H72
13																												
14	H72						M60		M60		M60					M61		M61		M61								H72
15							M60		M60		M60					M61		M61		M61								
16	H72						M60		M60		M60					M61		M61		M61								H72
17							M60	M60	M60	M60	M60		M60			M61	M61	M61	M61	M61		M61						
18	H72						M60	M60	M60	M60	M60		M60			M61	M61	M61	M61	M61		M61						H72
19							M60	M60	M60	M60	M60	M60				M61	M61	M61	M61	M61	M61							
20	H72							M60	M60	M60	M60						M61	M61	M61	M61								H72

2人で，ジャマしたり協力したりして，遊んでね

11 あるけあるけゲーム

ゲームの遊び方

1人で遊ぶのって，なんだかツマラナイ…。**2人で一緒に遊べたら**いいのにナ。

そんな風に思ったら，このゲームを遊んでみてネ。2人で遊ぶと，ゲームは**とてもとても面白くなる**んです。

カベとかの障害物にぶつかっちゃうと負けなんですネ
だから，相手を上手に囲んじゃえばいいの。

ゲームが始まりまーす。どうなるかな？

うー，残念だけど負けちゃった！

●コントローラⅠ

●コントローラⅡ

コントローラⅠ，コントローラⅡで，それぞれ動かします

遊ぶ前に，BGグラフィックを入れて，それからBASICプログラムを入力してネ。

このゲームは，2人で遊びます。ゲームがスタートした時には，テレビ画面の真ん中に2つ四角いキャラクタがあるの。左のキャラクタはコントローラⅠで，右のキャラクタはコントローラⅡで動かすんですー。障害物にぶつかっちゃった方が，負けですー。2人で遊ぶと，うん，**すっごく面白いんですっ！**

```
100  '●● GAME/11(チアキ+カナエ) ●●
110  VIEW:SPRITE OFF
120  PALETB 0,13,48,33,22
130  X=13:Y=10:M=14:N=10
140  VX=-1:VY=0:VM=1:VN=0
200  LOCATE 0,21:PRINT" *SCOR
E= 0     HISCORE=";H;"   *";
210  PAUSE100:PLAY"#D0"
220  LOCATE X,Y:PRINT CHR$(20
9);
230  LOCATE M,N:PRINT CHR$(19
5);
240  '
300  '●メインルーチン●
310  I=STICK(0)
320  IF I=2 THEN VX=-1:VY=0
330  IF I=1 THEN VX= 1:VY=0
340  IF I=8 THEN VY=-1:VX=0
350  IF I=4 THEN VY= 1:VX=0
360  I=STICK(0)
370  IF I=2 THEN VM=-1:VN=0
380  IF I=1 THEN VM= 1:VN=0
390  IF I=8 THEN VN=-1:VM=0
400  IF I=4 THEN VN= 1:VM=0
410  P$=SCR$(X+VX,Y+VY)
420  IF P$<>" " THEN1000
430  P$=SCR$(M+VM,N+VN)
440  IF P$<>" " THEN1100
450  X=X+VX
460  Y=Y+VY
470  M=M+VM
480  N=N+VN
490  LOCATE X,Y:PRINT CHR$(20
9);
500  LOCATE M,N:PRINT CHR$(19
5);
510  PLAY"A0":GOTO300
520  '
1000  '●ケ゛ームオーハ゛ー1●
1010  LOCATE 2,10:PRINT"** コン
トローラー=1 GAME OVER **";
1020  LOCATE 2,12:PRINT"サ゛ンネン
テ゛シタ. スコシ,カ゛ンハ゛ッテクタ゛サイネ!";
1030  INPUT"",A:GOTO110
1040  '
1100  '●ケ゛ームオーハ゛ー2●
1110  LOCATE 2,10:PRINT"** コン
トローラー=2 GAME OVER **";
1120  LOCATE 2,12:PRINT"サ゛ンネン
テ゛シタ. スコシ,カ゛ンハ゛ッテクタ゛サイネ!";
1130  INPUT"",A:GOTO110
1140  '
```

コントローラⅠの動きを調べてます。

コントローラⅡの動きを調べてます。

コントローラⅠを使っている人，ゲームオーバー

コントローラⅡを使っている人，ゲームオーバー

BGグラフィック画面11

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
0	L41	I61	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	I71	L41
1	L41	J21																									J21	L41
2	L41	J21																									J21	L41
3	L41	J21																									J21	L41
4	L41	J21																									J21	L41
5	L41	J21																									J21	L41
6	L41	J21																									J21	L41
7	L41	J21																									J21	L41
8	L41	J21																									J21	L41
9	L41	J21																									J21	L41
10	L41	J21												H10	F30												J21	L41
11	L41	J21																									J21	L41
12	L41	J21																									J21	L41
13	L41	J21																									J21	L41
14	L41	J21																									J21	L41
15	L41	J21																									J21	L41
16	L41	J21																									J21	L41
17	L41	J21																									J21	L41
18	L41	J21																									J21	L41
19	L41	J01	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J11	L41
20	L51	L61	L61	L61	L61	L61	L61	L61	L61	L61	L61	L61	L61	L61	L61	L61	L61	L61	L61	L61	L61	L61	L61	L61	L61	L61	L61	M01

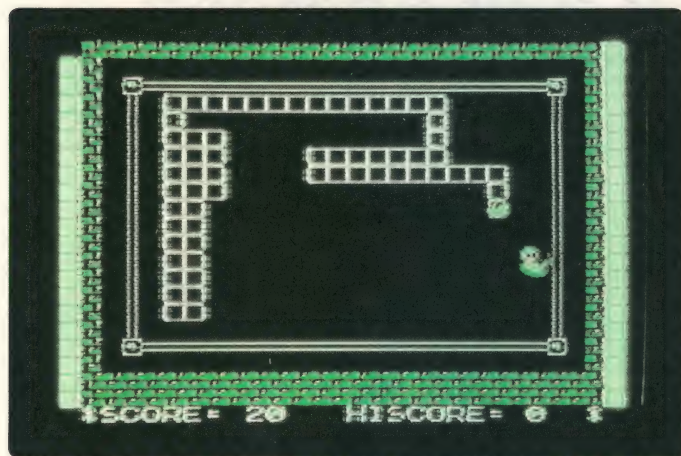
エィリアンをヘビを食べると成長するんです

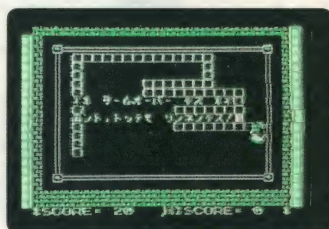
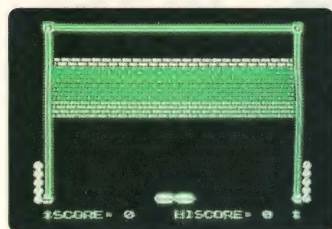
12 ハッスルしましょう

ゲームの遊び方

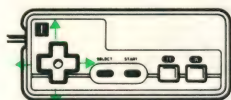
単純で面白いゲームを遊ぶのなら，コレガ一番のおすすめ。はじめはネ，カンタンなゲームなの。でも，それがヘビの長さがながくなると，**ぐんぐーん**と面白くなっちゃう。

ムムッ，なかなかやるな！，のゲームなんですー。

へビくんのカラダ，すいぶん長くなっちゃったー。こうなると，もう，動きにくくって，困っちゃうの

スタートしたばっかりだとこんなに短かいのにー

残念ながら，ゲームオーバーなんです。

ゲームを遊ぶ前に，BGグラフィックを入れて，それから**BASICプログラムを入力**してネ。

このゲームは，エイリアンをヘビくんがパックンて食べるんです。んっ　エイリアンをヘビくんが食べるなんて，なんだか**ヘン**？

でね，エイリアンを食べると，すこしだけ成長するんだって…。どんどん成長していって，だんだん動きづらくなっちゃうの。カベとか自分のカラダにぶつかるとゲームオーバーだもん。

コントローラで，ヘビを動かします。

```
100  '●● GAME/12(チアキ+カナエ) ●●
110  DIM X(150),Y(150)
120  CGSET1,0:VIEW:SPRITE ON
130  X=10:Y=10:R=0:L=9:F=-1
140  S=0:VX=-1:VY=0
150  DEF SPRITE 0,(3,1,0,0,0)
="チツテト"
160  LOCATE 0,21:PRINT" *SCOR
E= 0     HISCORE=";H;"   *";
170  PALETB 0,13,48,41,2
180  PALETB 2,13,48,33,22
200  FOR I=0 TO 9
210  LOCATE I+10,10:PRINT CHR
$(176);
220  X(I)=I+10:Y(I)=10
230  NEXT
240  LOCATE 10,10:PRINT CHR$(
207);
250  PAUSE150:PLAY"A2R0B2"
260  '
300  '●メインルーチン●
310  IF F GOSUB1000
350  I=STICK(0)
360  IF I>0 GOSUB1100
400  P$=SCR$(X+VX,Y+VY)
410  IF P$<>" " GOSUB1200
420  LOCATE X,Y:PRINT CHR$(17
6);←───ヘビさんの頭を表示します
430  X=X+VX
440  Y=Y+VY
450  LOCATE X,Y:PRINT CHR$(20
7);
460  R=R-1
470  IF R<0 THEN R=150
480  X(R)=X:Y(R)=Y
490  LOCATE X(L),Y(L):PRINT"
";←───ヘビさんのシッポを消します
500  L=L-1
510  IF L<0 THEN L=150
520  GOTO300
530  '
1000  '●サブルーチン1●
1010  I=RND(20)+4
1020  J=RND(14)+3
1030  P$=SCR$(I,J)
1040  IF P$<>" " THEN1000
1050  LOCATE I,J:PRINT".";
1060  SPRITE 0,I*8+12,J*8+18
1070  F=0
1080  RETURN
1090  '
1100  '●サブルーチン2●
1110  IF I=2 THEN VX=-1:VY=0
1120  IF I=1 THEN VX= 1:VY=0
1130  IF I=8 THEN VY=-1:VX=0
1140  IF I=4 THEN VY= 1:VX=0
1150  RETURN
1160  '
1200  '●サブルーチン3●
1210  IF P$ =". " THEN1300
1220  IF P$<>". " THEN1400
1230  RETURN
```

100行から180行までは，画面の設定と変数の設定をしてます。

200行から240行では，ヘビさんの設定と表示をしてます。

350行から510行で，ヘビさんを動かしてます。変数X，Yはヘビさんの頭の位置です。変数X(L)，Y(L)はしっぽの位置です。

サブルーチン1は，ファイアーボールを表示するとこです。

サブルーチン2は，コントローラの動きを変数VX，VYに入れるとこです。

サブルーチン3は，ぶつかった物を調べるルーチンです

サブルーチン4は，ファイアーボールを取った時のためのことです。

```
1240 '
1300 '●ｻﾌﾞﾙｰﾁﾝ4●
1310 F=-1:S=S+10
1320 LOCATE 8,21:PRINT S;
1330 PLAY"C1F1C1B1"
1340 L=L+1
1350 IF L>150 THEN L=0
1360 RETURN
1370 '
1400 '●ｻﾌﾞﾙｰﾁﾝ5●
1410 LOCATE 5, 8:PRINT"** ｹﾞｰﾑｵｰﾊﾞｰ ﾃﾞｽ **";
1420 LOCATE 5,10:PRINT"ﾎﾝﾄ,ﾄｯﾃﾓ ｻﾞﾝﾈﾝﾃﾞｽ!";
1430 IF S>H THEN H=S
1440 INPUT"",A:GOTO120
```

BGグラフィック画面12

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
0	D72	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	D52
1	D72	F32																									F32	D52
2	D72	F32		D40	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	D40		F32	D52
3	D72	F32		J20																					J20		F32	D52
4	D72	F32		J20																					J20		F32	D52
5	D72	F32		J20																					J20		F32	D52
6	D72	F32		J20																					J20		F32	D52
7	D72	F32		J20																					J20		F32	D52
8	D72	F32		J20																					J20		F32	D52
9	D72	F32		J20																					J20		F32	D52
10	D72	F32		J20																					J20		F32	D52
11	D72	F32		J20																					J20		F32	D52
12	D72	F32		J20																					J20		F32	D52
13	D72	F32		J20																					J20		F32	D52
14	D72	F32		J20																					J20		F32	D52
15	D72	F32		J20																					J20		F32	D52
16	D72	F32		J20																					J20		F32	D52
17	D72	F32		D40	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30			F32	D52
18	D72	F32																									F32	D52
19	D72	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	D52
20	D72	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	D52

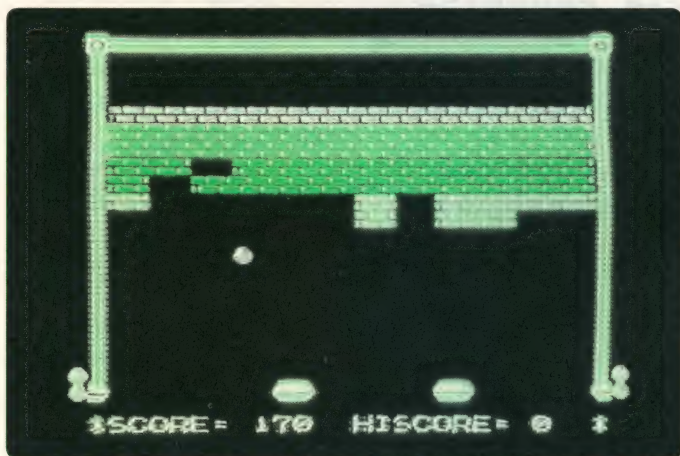
両手を使って遊ぶブロックくずしです

13 ブロックくずし・スペシャル

ゲームの遊び方

人間の**右手**と**左手**って，なかなか上手に協力してくれないの。たとえば，お料理をつくる時，右手は**キャベツ**を切って，左手は**目玉焼**なんて，できないもん。できたら，便利，なのにネ。

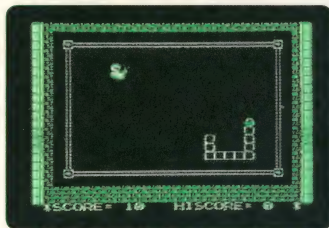
そんなわけで，両手で遊ぶゲーム，ですー。

テレビ画面の下の方には，2つスティックがあります。これをコントローラⅠとⅡで動かします。
画面の上の方にあるのは，ブロックです。これをぜーんぶ消しちゃいますー。

スタートした時の画面です。

ボール5コ全部使っちゃったから，ゲームオーバーです。

ゲームを遊ぶ前に，BGグラフィックを入れて，それからBASICプログラムを入れてネ。

このゲームは，コントローラⅠとⅡの**両方を使う**ブロックくずしなんですー。どう，なかなかスゴイでしょ？

ボールをスティックではね返して，ブロックをくずしてね。ボール5コで，全部のブロックを消しちゃえばいいの。あきたら，BGグラフィックの**ブロックを変えて**遊びましょ！

コントローラⅠで左のスティックを動かします。コントローラⅡで右のスティックを動かします。
ゲームオーバーから，もう1度遊びたい時は，RETURNを押してくださいネ。

```
100  '●● GAME/13(チアキ+カナエ) ●●
120  VIEW:SPRITE OFF
130  PALETB 0,13,48,33,25
140  PALETB 1,13,48,36,20
150  PALETB 2,13,48,37,21
160  PALETB 3,13,48,44,28
180  O=0:F=-1:S=0:B=5
190  L=12:R=14:C$="  "
200  LOCATE 0,22:PRINT" *SCOR
E=  0      HISCORE=";H;"    *";
210  PAUSE150
220  '
300  '●メインルーチン●
310  IF F GOSUB1000
350  LOCATE X,Y:PRINT C$;
360  P$=SCR$(X+VX,Y+VY):C$="
"
370  IF P$<>"  " GOTO1100
380  IF SCR$(X,Y+VY)=CHR$(195
) GOSUB1200
400  X=X+VX
410  Y=Y+VY
420  LOCATE X,Y:PRINT CHR$(20
7);
430  IF Y=21 THEN1500
450  I=STICK(0)
460  J=STICK(1)
470  IF I>0 GOSUB1300:GOTO500

480  IF J>0 GOSUB1400:GOTO500
490  PAUSE2
500  GOTO300
510  '
1000  '●サブルーチン1●
1010  X=RND(24)+2
1020  VX=RND(2)*2-1
1030  Y=19
1040  VY=-1
1050  F=0
1060  PLAY"T1M1V0C0#C0C0#C0C0
"
1070  LOCATE X,Y:PRINT CHR$(2
7);
1080  RETURN
1090  '
1100  '●サブルーチン2●
1110  IF P$=CHR$(195) THEN C$
=CHR$(195(:GOTO380
1120  IF P$=CHR$(226) THEN VX
=-VX:GOSUB1600
1140  IF P$=CHR$(227) THEN VY
=-VY:GOSUB1600
1150  IF P$=CHR$(196) THEN VX
=-1:VY=-1:GOSUB1600
1160  IF P$=CHR$(198) THEN VX
= 1:VY=-1:GOSUB1600
1170  GOTO360
1180  '
1200  '●サブルーチン3●
1210  LOCATE X/2*2,Y-1:PRINT"
  ";
1220  VY=-VY:S=S+10:C$="  "
```

千秋と香奈枝が作りました。

ボールを消します

何にぶつかったの？

変数Oは，ゲームオーバーになったかどうか。変数は，ボールの数です。

310行では，Fが0じゃない時，ボールの位置を決めてます

350から430行では，ボールを動かしてます。

450から480行では，左右のスティックを動かしてます。

サブルーチン1では，ボールの位置を決めてます。

サブルーチン2では，ボールが何かにぶつかった処理をしてます。

サブルーチン3では，スコアに加算してます。

サブルーチン4では，左のスティックを動かしてます。

サブルーチン5では，右のスティックを動かしてます。

もう1度遊びたい人は，RETURNを押してネ

```
1230  PLAY"T5B1"
1240  LOCATE 8,22:PRINT S;
1260  RETURN
1270  '
1300  '●サブルーチン4●
1310  LOCATE L,20:PRINT"   ";
1320  L=L-(I=2)*(L<>2)+(I=1)*
(L<>12)←──コントローラ入力
1330  LOCATE L,20:PRINT CHR$(
196);CHR$(198);
1340  RETURN
1350  '
1400  '●サブルーチン5●
1410  LOCATE R,20:PRINT"   ";
1420  R=R-(J=2)*(R<>14)+(J=1)
*(R<>24)←──コントローラ入力
1430  LOCATE R,20:PRINT CHR$(
196);CHR$(198);
1440  RETURN
1450  '
1500  '●サブルーチン6●
1510  PLAY"T3M1V1D0F0D0F0D0C1
"
1520  LOCATE 0,21-B:PRINT" ";
1530  LOCATE27,21-B:PRINT" ";
1540  LOCATE X,Y:PRINT" ";
1550  B=B-1:F=-1:C$=" "
1560  IF B=0 THEN1700
1570  PAUSE150
1580  GOTO300
1590  '
1600  '●サブルーチン7●
1610  LOCATE X,Y:PRINT CHR$(2
55);
1620  PLAY"T1M1V0C0#C0"
1630  LOCATE X,Y:PRINT" ";
1640  RETURN
1650  '
1700  '●ゲームオーバー●
1710  LOCATE 5,10:PRINT" ** G
AME OVER **";
1720  LOCATE 5,12:PRINT"ザンネン
デスガ ゲームオーバー!";
1730  IF S>H THEN H=S
1740  INPUT"",A:GOTO120
```

BGグラフィック画面13

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
0		D40	J20	J20	J20	J20	J20	J20	J20	J20	J20	J20	J20	J20	J20	J20	J20	J20	J20	J20	J20	J20	J20	J20	J20	J20	D40	
1		J30																									J30	
2		J30																									J30	
3		J30																									J30	
4		J30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	J30	
5		J30	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	J30	
6		J30	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	J30	
7		J30	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	J30	
8		J30	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	J30	
9		J30	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	J30	
10		J30	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	J30	
11		J30																									J30	
12		J30																									J30	
13		J30																									J30	
14		J30																									J30	
15	G70	J30																									J30	
16	G70	J30																									J30	G70
17	G70	J30																									J30	G70
18	G70	J30																									J30	G70
19	G70	J30																									J30	G70
20	G70	F60											F40	F60	F40	F60											F40	G70

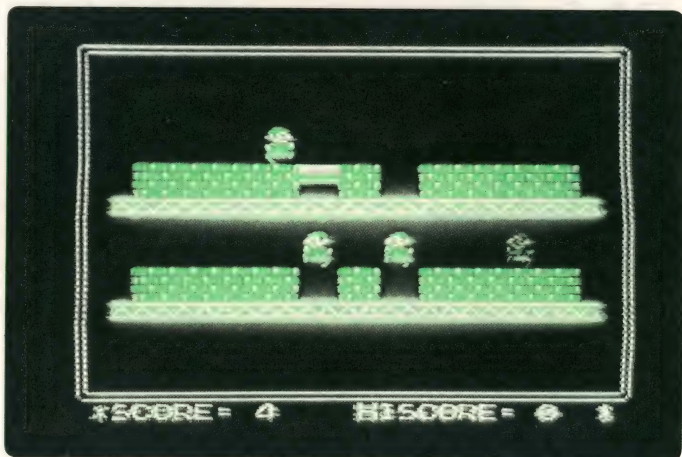
道路のマンホールに人が落ちないように！

14 マンホール大作戦

ゲームの遊び方

道路の**マンホールのフタ**が，はずれっぱなしになってます。ワァ，**これは危険**，ですよネ。もしも，歩いてる人が落っこちちゃったら…。想像しただけで，**ヒヤアセ…**。

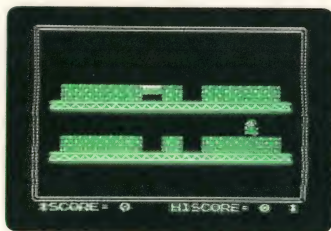
落っこちちゃわないよーに，助けてあげてくださーい。

テレビ画面には，道路が2本あります。それぞれに2つずつ穴ボコがあいてます。ネ，マンホールなんです。このマンホールに人が落ちないようにするのが，このゲームですー。

マンホールに落っこちないように，助けてあげてネ！

落っこちちゃうと，ゲームオーバーなんです。

ゲームを遊ぶ前に，BGグラフィックを入れて，BASICプログラムを入力してネ。

このゲームは，道路を歩いてる人がマンホールに落っこちないようにするんです。落っこちないようにするには，**マンホールにフタをするの**。簡単なんです。

1人落ちないようにすると1点です。歩く人が，だんだん増えてくから，なかなかタイヘンですー。よいしょ…。

マンホールのフタは，コントローラIIのBボタンで右上に，Aボタンで右上に動きます。それから，コントローラIのBボタンで左下に，Aボタンで右下に動きます。

もう1度遊びたい人は，RETURNを押してね。

```
100  '●● GAME/14(チアキ+カナエ) ●●
110  DIM X(7),Y(7),V(7)
120  CGSET1,0:VIEW:SPRITE ON
130  PALETB 1,13,48,41,24
140  S=0:X=11:Y=7:U=0:D=0
200  FOR I=0 TO 7
210   V(I)=0:SPRITE I
220  NEXT
230  LOCATE 0,21:PRINT" *SCOR
E= 0     HISCORE=";H;"   *";
240  '
300  '●メインルーチン●
310  FOR I=0 TO 7
320  IF V(I) =0 GOSUB1000
330  IF V(I)<>0 GOSUB1300
340  PAUSE1
350  M=STRIG(1):N=STRIG(0)
360  IF M+N>0 GOSUB1400
370  IF V(I)<>0 GOSUB1600
380  NEXT
500  U=U+(U>0):D=D+(D>0)
510  GOTO300
520  '
1000  '●サフ゛ルーチン1●
1010  J=RND(30-S/3):J=-J*(S<9
0)
1020  IF J RETURN
1030  R=RND(2)
1040  IF R=0 GOTO1100  ← 上のマリオに設定します
1050  IF R=1 GOTO1200  ← 下のマリオに設定します
1060  RETURN
1070  '
1100  '●サフ゛ルーチン2●
1110  IF U>0 THEN RETURN
1120  X(I)= 5:Y(I)= 5:V(I)= 2
1130  U=2
1140  DEF SPRITE I,(0,1,0,1,0
)=CHR$(1)+CHR$(0)+CHR$(3)+CH
R$(2)
1150  RETURN
1160  '
1200  '●サフ゛ルーチン3●
1210  IF D>0 THEN RETURN
1220  X(I)=21:Y(I)=11:V(I)=-2
1230  D=2
1240  DEF SPRITE I,/0,1,0,0,0
)=CHR$(0)+CHR$(1)+CHR$(2)+CH
R$(3)
1250  RETURN
1260  '
1300  '●サフ゛ルーチン4●
1310  IF X(I)=3 OR X(I)=23 TH
EN SPRITE I:V(I)=0:RETURN
1320  SPRITE I,X(I)*8+16,Y(I)
*8+22  ← マリオを表示してま～す
1350  RETURN
1360  '
1400  '●サフ゛ルーチン5●
1410  LOCATE X,Y:PRINT"  ";
1420  IF M=4 THEN X=11:Y= 7
1430  IF M=8 THEN X=15:Y= 7
1440  IF N=4 THEN X=11:Y=13
```

320行では，マリオが表示されてないならサブルーチン1へ。表示してたらサブルーチン2へ

サブルーチン1は，たまにマリオを表示するようにしてます

サブルーチン2は，上のマリオを表示したいときに使ってます

サブルーチン3は，下のマリオを表示したいときに使ってます

サブルーチン4は，マリオを動かしてます

サブルーチン5は，コントローラの動きによって，フタを移動させてます

```
1450  IF  N=8  THEN  X=15:Y=13
1460  LOCATE  X,Y:PRINT  CHR$(1
97);CHR$(197);
1470  RETURN
1480  '
1500  '●サブルーチン6●
1510  P$=SCR$(X(I),Y(I)+2)
1520  IF  P$<>CHR$(197)  GOTO17
0
1530  S=S+1
1540  LOCATE  0,2:PRINT  S;
1550  PLAY"T2C0A0"
1560  RETURN
1570  '
1600  '●サブルーチン7●
1610  IF  X(I)=11  OR  X(I)=15  G
OSUB1500
1620  X(I)=X(I)+V(I)
1630  RETURN
1640  '
1700  '●ゲームオーバー●
1710  LOCATE  5,2:PRINT"**GAME
 OVERデス**";
1720  LOCATE  5,4:PRINT"ザンネンデ
シタネ.ガンバッテ!";
1730  IF  S>H  THEN  H=S
1740  INPUT"",A:GOTO120
```

BGグラフィック画面14

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
0	I61	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	I71
1	J21																											J21
2	J21																											J21
3	J21																											J21
4	J21																											J21
5	J21																											J21
6	J21																											J21
7	J21			F32	F32	F32	F32	F32	F32	F32	F32	F52	F52	F32	F32			F32	F32	F32	F32	F32	F32	F32	F32			J21
8	J21			F32	F32	F32	F32	F32	F32	F32	F32			F32	F32			F32	F32	F32	F32	F32	F32	F32	F32			J21
9	J21		H00	H00	H00	H00	H00	H00	H00	H00	H00	H00	H00	H00	H00	H00	H00	H00	H00	H00	H00	H00	H00	H00	H00	H00		J21
10	J21																											J21
11	J21																											J21
12	J21																											J21
13	J21			F32	F32	F32	F32	F32	F32	F32	F32			F32	F32			F32	F32	F32	F32	F32	F32	F32	F32			J21
14	J21			F32	F32	F32	F32	F32	F32	F32	F32			F32	F32			F32	F32	F32	F32	F32	F32	F32	F32			J21
15	J21		H00	H00	H00	H00	H00	H00	H00	H00	H00	H00	H00	H00	H00	H00	H00	H00	H00	H00	H00	H00	H00	H00	H00	H00		J21
16	J21																											J21
17	J21																											J21
18	J21																											J21
19	J21																											J21
20	J01	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J31	J11

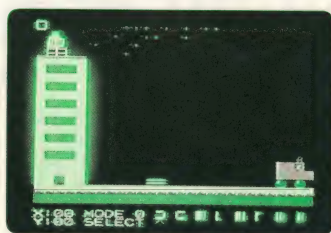
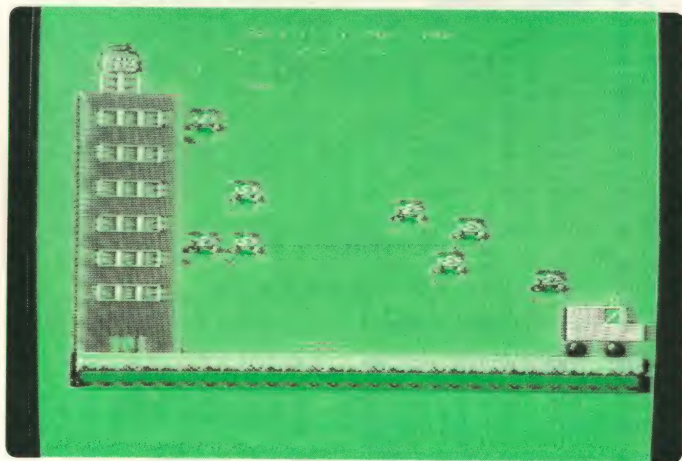
ビル火災がおきました。サァ，どうする！

15 ファイヤー救助隊

ゲームの遊び方

ビルに，**火事が**起きちゃったんですー！　逃げられなかった人が，つぎつぎと**飛び降り**はじめました。ワ，どうしよー!!

あなたは『レスキュー隊員』になって，この飛び降りる人をトランポリンで助けてあげてネ。

テレビ画面の左には，火事があったビルがあります。ビルの上から，人が飛び降りてくるんです。
画面の下には，救急車とトランポリンがあります。
でも，助けられないとゲームオーバーになっちゃうの。たいへんなんです。

ゲームを遊ぶ前に，BGグラフィックを入れて。それから，BASICプログラムを入力してネ。

このゲームは，**火事のあったビルから**飛び降りる人を助けるんです。次々に飛び降りてくるから，もうタイヘンなのー。

トランポリンを上手に動かして，**救急車まで運んでね**。でも，助けられないと…。その場で『救急隊員』失格です。

だんだんと，人が増えてくるから難しいですヨ。

コントローラで左右に動きます。

```
100  '●● GAME/15(チアキ+カナエ) ●●
110  DIM X(24),Y(24),M(7)
120  CGSET1,0:VIEW:SPRITE ON
130  PALETB 0,12,48,0,13
140  PALETB 2,13,48,37,37
150  X=11:Y=0:F=16:FV=2
200  FOR I=5 TO 16 STEP2
210  LOCATE 1,I:PRINT CHR$(1
76);CHR$(176);CHR$(176);
220  NEXT
230  RESTORE
240  FOR I=0 TO 24
250  READ X(I),Y(I)
260  X(I)=X(I)*8+16
270  Y(I)=Y(I)*8+22
280  NEXT
300  DATA 3,2,5,3,5,5,5,8,5,1
2,6,16,7,12,7,9,8,6,9,4,10,6
310  DATA 10,9,10,12,11,16,12
,13,13,10,14,8,15,10,15,13
320  DATA 16,16,17,13,18,11,2
0,13,21,16,22,14
330  LOCATE 0,21:PRINT" *SCOR
E= 0     HISCORE=";H;"   *";
340  FOR I=0 TO 7
350  M(I)=255 SPRITE I
360  NEXT
370  GOSUB1400
380  '
400  '●メインルーチン●
410  FOR I=0 TO 7
420  IF M(I)=255 GOSUB1000
430  IF M(I)<255 GOSUB1100
440  'GOSUB1400
450  C=STICK(0)
460  IF C>0 GOSUB1300
470  IF M(I)<255 GOSUB1500
500  NEXT
510  Y=Y+(Y>0)
520  GOTO400
530  '
1000  '●サブルーチン1●
1010  J=RND(50-S/3):J=-J*(S<9
0)
1020  IF J RETURN
1030  IF Y>0 THEN RETURN
1040  M(I)=0:Y=5
1050  RETURN
1060  '
1100  '●サブルーチン2●
1110  DEF SPRITE I,(0,1,0,1,0
)=CHR$(1)+CHR$(0)+CHR$(3)+CH
R$(2)
1120  SPRITE I,X(M(I)),Y(M(I)
)
1130  M(I)=M(I)+1
1140  IF M(I)=25 GOTO1200
1150  RETURN
1160  '
1200  '●サブルーチン3●
1210  SPRITE I
1220  S=S+1:M(I)=255
```

(440行) コレを取ると，ファイヤーが動きまわります

(1010行) スコアによって，難しさが変わります

200行から280行では，マリオの動く位置を配列X(I)とY(I)に読み込んでます。300行から320行は，そのためのデータです。340行から360行は，マリオのクリアをしてます。

420行は，マリオが表示されてないならサブルーチン1へ。430行は，表示されてたらサブルーチン2へ。440行のREM文を取るとファイヤーが動きまわります。

サブルーチン1は，たまにマリオを表示するようにするルーチンです。

サブルーチン2は，マリオを動かして表示するルーチンです。

サブルーチン3は，スコア表示してます

サブルーチン4は，コントローラの動きの通りに…マトを動かします。

サブルーチン5は，ファイヤを動かすルーチンです。

```
1230  LOCATE 8,21:PRINT S;
1240  PLAY"T1#C0B0C0B0C0B0";
1250  RETURN
1260  '
1300  '●サフ゛ルーチン4●
1310  LOCATE X,18:PRINT"   ";
1320  IF C=2 AND X> 6 THEN X=
X-5
1330  IF C=1 AND X<21 THEN X=
X+5
1340  LOCATE X,18:PRINT CHR$(
227);CHR$(227);←マットを表示します
1350  RETURN
1360  '
1400  '●サフ゛ルーチン5●
1410  IF FV= 2 THEN DEF MOVE(
0)=SPRITE(5,3,1,1,0,3)
1420  IF FV=-2 THEN DEF MOVE(
0)=SPRITE(5,7,1,1,0,3)
1430  POSITION 0,F,46
1440  MOVE 0
1450  F=F+FV
1460  IF F=48 THEN FV=-2
1470  IF F=16 THEN FV= 2
1480  RETURN
1490  '
1500  '●サフ゛ルーチン6●                マリオはマットの上？
1510  IF Y(M(I)-1)<>150 RETUR
N
1520  IF X(M(I)-1)<>X*8+16 GO
TO1600
1530  PLAY"T1F2G2"
1540  RETURN
1550  '
1600  '●ケ゛ームオーハ゛ー●
1610  LOCATE 5,2:PRINT"**GAME
 OVERテ゛ス**";
1620  LOCATE 5,4:PRINT"サ゛ンネンテ゛
シタネ.カ゛ンハ゛ッテ!";
1630  IF S>H THEN H=S
1640  INPUT"",A:GOTO120
```

BGグラフィック画面15

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
0											G50		G50	G50	G50	G50	G50											
1		M10	M30				G50		G50		G50	G50	G50															
2		M20	M40		G50	G50		G50		G50		G50																
3		H20	H20		G50		G50																					
4	B70	B70	B70	B70	B70																							
5	M60				M60																							
6	M60	M60	M60	M60	M60																							
7	M60				M60																							
8	M60	M60	M60	M60	M60																							
9	M60				M60																							
10	M60	M60	M60	M60	M60																							
11	M60				M60																							
12	M60	M60	M60	M60	M60																							
13	M60				M60																							
14	M60	M60	M60	M60	M60																							
15	M60				M60																						J42	
16	M60	M60	M60	M60	M60																				M50	M50	L72	
17	M60	M60	M60	M60	M60																				M50	M50	M50	M50
18	M60	M60	K60	M60	M60							J30	J30												G70		G70	
19	I13	I13	I13	I13	I13	I13	I13	I13	I13	I13	I13	I13	I13	I13	I13	I13	I13	I13	I13	I13	I13	I13	I13	I13	I13	I13	I13	I13
20	I43	I43	I43	I43	I43	I43	I43	I43	I43	I43	I43	I43	I43	I43	I43	I43	I43	I43	I43	I43	I43	I43	I43	I43	I43	I43	I43	I43

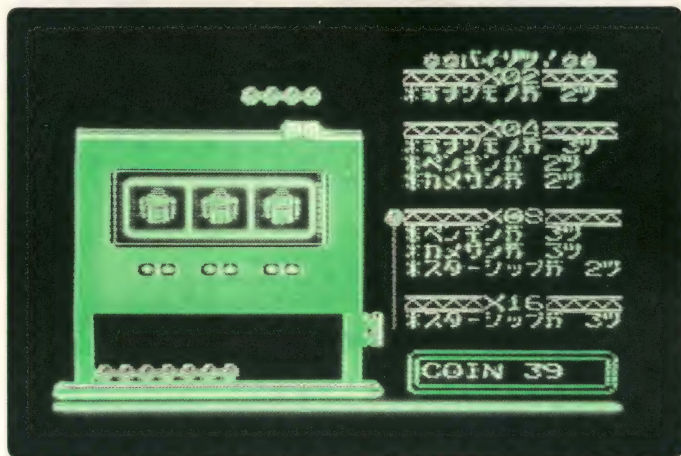
コインがぞろぞろ，本格的なスロットマシーン。

16 スロットマシーン

ゲームの遊び方

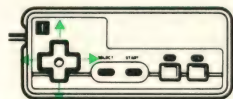
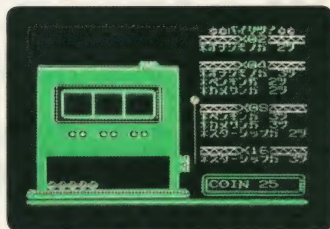
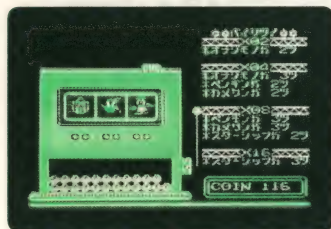
健全なブロットマシーンです。

このスロットマシーンは，とっても良くできてるナー，と思います。だってネ，役だって**たくさんあるし**，**レバー**は引けるしそれに，**コイン**だって，ドコドコ出てくるんだもん。

入れたコインの枚数は，スロットマシーンの上に出ます。
スロットマシーンの受け皿にあるコインは，あなたの持ってる枚数です。
表示されてる1コは，5枚ぶんです。

わー，こんなに増えちゃいましたー。すごいっ！

ゲームオーバー！

ゲームを遊ぶ前に，BGグラフィックを入れて。それから，BASICプログラムを入力してネ。

まず，賭けるコインの数を決めます。Bボタンを押して，コインを入れてね。7枚まで入るの。で，Aボタンを押せば，レバーが引かれて，クルクル回転するんです。Aボタンを押せば，回転は止まります。こうして，止まったマークによって，コインの数が増えたり減ったりします。そうそう，コインは最初25枚ですー。

Bボタンは，コインの枚数を決めます。Aボタンは，レバーを引いたり，回転を止めたりします。

```
100  '●● GAME/16(チアキ+カナエ) ●●
110  DIM A$(7),B(7),C(7),M(2)
120  VIEW:SPRITE ON
130  PALETB 0,13,48,37,33
150  PALETB 2,13,48,41,25
160  PALETB 3,13,48,48,39
162  PALETS 0,13,48,22,1
163  PALETS 1,13,16,0,1
165  PALETS 2,13,48,39,22
170  S=25
180  M(0)=0:M(1)=0:M(2)=0
200  A$(0)="89:;":B(0)=0
210  A$(1)="ケコサシ":B(1)=0
220  A$(2)="ゾダヂヅ":B(2)=1
230  A$(3)="XYZ「":B(3)=0
240  A$(4)="ピプペポ":B(4)=0
250  A$(5)="チツテト":B(5)=2
260  A$(6)="@ABCD":B(6)=0
270  A$(7)=CHR$(184)+CHR$(185
)+CHR$(186)+CHR$(187):B(7)=2
280  '
290  '●メインルーチン●
300  I$="             "
305  IF S<0 GOTO1200
310  FOR I=18TO15STEP-1
320  LOCATE2,I:PRINT I$;
330  NEXT
335  LOCATE22,18:PRINT S;"  ";
340  I$="":J=18
350  FOR I=1 TO S/5
360  I$=I$+CHR$(207)
370  IF LEN(I$)=12 LOCATE2,J:
PRINT I$:I$="":J=J-1
380  NEXT
390  LOCATE2,J:PRINT I$
400  C=0:LOCATE6,2:PRINT"
  "
410  I=STRIG(0)
420  IF I=4 GOSUB1000
430  IF I=8 OR C=7 GOTO460
440  GOTO410
450  '
460  IF C=0 GOTO410
470  S=S-C:PLAY"T1C0B0C0B0C0B
0C0B0C0B0C0B0C0B0"
480  FOR I=8TO13
490  LOCATE16,I:PRINT"  ";
500  LOCATE16,I+1:PRINT CHR$(
207);:PAUSE20
510  NEXT
520  I=0
530  GOSUB1100:PAUSE2
540  IF STRIG(0)<>8 GOTO530
550  PLAY"T1C1D1F1"
560  FORJ=1 TO 10-(RND(3)-1)*
(RND(2)=0)
570  GOSUB1100:PAUSEJ
580  NEXT
590  I=I+1:IF I<3 GOTO530
600  M=0
610  FOR I=0TO7
```

← コインの量が決ったらー（400行）

← ルーレットを回転させるルーチン（530行）

130行から165行までは，パレットの設定をしてます。180行から270行では，ルーレットのマークを決めてます。

300行から400行では，コインの量を表示してます。

410行から440行では，入れるコインの量を決めてます。

460行から510行では，スロットマシーンのレバーを引くように表示してます。

520行から590行では，ルーレットを回転させてます。変数Iは，回転するルーレットの値です。

600行から710行では，マークの状態によってスコアを決めてます。変数Mは，その倍数。変数Cは，入れたコインの枚数。

```
620  C(I)=Ø
630  NEXT
640  FOR I=ØTO2
650  C(M(I))=C(M(I))+1
660  NEXT
670  FOR I=ØTO7
680  IF C(I)>1 M=C(I)
690  NEXT
700  IF C(2)>1 OR C(Ø)>1 M=M*
2
710  IF C(5)>1 M=M*4
```

750行から780行では，スロットレバーを元の状態にもどしてます。それから，790行では，スコアに加算してます。

```
750  FOR I=15TOSTEP-1
760  LOCATE16,I:PRINT CHR$(23
8);
770  NEXT
780  LOCATE16,9:PRINT CHR$(2Ø
7);
790  S=S+C*M
800  IF M>Ø PLAY"T1C3F3C3B3"
810  GOTO3ØØ
820  '
```

サブルーチン1では，コインを入れるのをしています。

```
1ØØØ  '●サフ゛ルーチン1●
1Ø1Ø  LOCATE12-C,2:PRINT CHR$
(2Ø7);
1Ø2Ø  C=C+1
1Ø3Ø  PLAY"T1B5D5"
1Ø4Ø  RETURN
1Ø5Ø  '
```

サブルーチン2では，ルーレットを回すのをしてます。

```
11ØØ  '●サフ゛ルーチン2●
111Ø  M=M(I)
112Ø  M(I)=-(M+1)*(M<7)
113Ø  DEF SPRITE I,(B(M),1,Ø,
Ø,Ø)=A$(M)
114Ø  SPRITE I,48+I*24,86
115Ø  RETURN
116Ø  '
12ØØ  '●ケ゛ームオーハ゛ー●
121Ø  LOCATE3,16:PRINT"*GAMEO
VER*";
122Ø  INPUT"",A:GOTO12Ø
```

BGグラフィック画面16

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
0																			H72	H72	バ	イ	リ	ツ	／	H72	H72	
1																		H02	H02	H02	H02	X	0	2	H02	H02	H02	H02
2							G73	G73	G73	G73	G73	G73	G73					*	オ	ナ	ジ	モ	ノ	ガ		2	ツ	
3																												
4		E40	E40	E40	E40	E40	E40	E40	E40	E40	E40	M20	M40	E40	E40			H02	H02	H02	H02	X	0	4	H02	H02	H02	H02
5		M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60			*	オ	ナ	ジ	モ	ノ	ガ		3	ツ	
6		M60	M60	B60	B60	B60	B60	B60	B60	B60	B60	B60	B60	M60	M60			*	ペ	ン	ギ	ン	ガ		2	ツ		
7		M60	K50	K70	K50	K50	K20	K50	K50	K20	K50	K50	L00	C60	M60			*	カ	メ	サ	ン	ガ		2	ツ		
8		M60	K50				K60			K60			K60	C60	M60		G72											
9		M60	K50				K60			K60			K60	C60	M60		K62	H02	H02	H02	H02	X	0	8	H02	H02	H02	H02
10		M60	K50	L10	K50	K50	K10	K50	K50	K10	K50	K50	L20	C60	M60		K62	*	ペ	ン	ギ	ン	ガ		3	ツ		
11		M60	M60	B70	B70	B70	B70	B70	B70	B70	B70	B70	B70	M60	M60		K62	*	カ	メ	サ	ン	ガ		3	ツ		
12		M60	M60	M60	D40	D40	M60	D40	D40	M60	D40	D40	M60	M60	M60		K62	*	ス	タ	ー	シ	ッ	プ	ガ		2	ツ
13		M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60		K62											
14		M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60		K62	H02	H02	H02	H02	X	1	6	H02	H02	H02	H02
15		M60													M60	D20	K62	*	ス	タ	ー	シ	ッ	プ	ガ		3	ツ
16		M60													M60	D20												
17		M60													M60		I60	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	I70
18		M60	G73	G73	G73	G73									M60		J20	C	O	I	N		2	5				J20
19	F40	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50	F60	J00	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30
20	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50

気球をゴールまで上手に運ぶゲームです

17 バルーン・ボンバー

ゲームの遊び方

ポコポコ揺れる『気球』に乗って，小旅行…。

うーん，障害物はあるし，ちゃんとした通り道はないし，なかなか気球で旅行するのも，**タイヘンなんですネ。**

でも，上手になったら，もっと遠くへ行ってみたいナ。

テレビ画面の右下にあるのは，気球です。左上にあるのは，ゴール。それから，動いてるブロックは障害物なんです。

ゴールインするのは，ハタの上に気球をのせてAボタンを押します。

このゲームを遊ぶ前に，BGグラフィックを入れて。それから，BASICプログラムを入力してネ。BGグラフィックは，あなたのレベルに合わせて選びますー。えと，練習用は，ゲームじゃなくって，いろんな難しいコースを練習するためにあるんですよお。

このゲームは，画面の右下にある気球を，左上のゴール地点まで運ぶんです。気球は，**コントローラで動きます。**

ゴールのハタの上に気球をのせて，で，Aボタンを押せばゴールインなんです。**ハタにぶつけちゃ**，だめなのー。

LEVEL１が成功したらLEVEL２っていうふうに，だんだんレベルを上げてってネ。でもお，LEVEL４はすーっごく難しいの。ん，でも，こればっかりは練習するしかないと思う…。

自分で好きなコースを，BGグラフィックで作ってみるのもいいですねえ。**うんうん。**

コントローラで気球を動かします　ゴールのハタの上に気球をのせたら，Aボタンを押してネ！

```
100  '●● GAME/17(チアキ+カナエ) ●●
110  DIM M(7),N(7)
120  CGSET1,2:VIEW:SPRITE ON
130  PALETB 0,13,48,33,25
140  S=0:D=995:C=0:L=2:R=6
150  X=24:Y=18:A=24:B=16
160  RESTORE
170  FOR I=0TO7
180  READ N(I)
190  NEXT
200  DATA 1,1,1,1,-1,-1,-1,-1
210  FOR I=0TO7
220  READ M(I)
230  NEXT
250  DATA 1,0,-1,0,-1,0,1,0
260  DEF SPRITE 0,(2,1,0,0,0)
=CHR$(64)+CHR$(65)+CHR$(66)+
CHR$(67)
270  LOCATE 0,21:PRINT" *SCOR
E= 0     HISCORE=";H;"    *";
280  '
300  '●メインルーチン●
310  I=STICK(0):VX=0:VY=0
320  IF I<>0 GOSUB1000
330  IF STRIG(0)=8 GOSUB1100
350  LOCATE X,Y  :PRINT" ";
360  LOCATE A,B+1:PRINT" ";
370  X=X+VX
380  Y=Y+VY
390  A=A+M(C)+VX
400  B=B+VY
410  P$=SCR$(X,Y)+SCR$(A,B)+S
CR$(A,B+1)
420  IF P$<>"   " GOTO1300
430  LOCATE X,Y  :PRINT CHR$(
254);
440  LOCATE A,B+1:PRINT CHR$(
181);
450  SPRITE 0,A*8+12,B*8+18
460  '
470  LOCATE L,R:PRINT" ";
480  L=L+N(C)
490  LOCATE L,R:PRINT CHR$(19
2);
500  C=-(C+1)*(C<7)
510  D=D-5
520  IF D=0 GOTO1300
530  LOCATE31,0:PRINT D;" ";
540  'PAUSE50
550  GOTO330
560  '
1000  '●サブルーチン1●
1010  VX=0:VY=0:S=S+1
1020  IF (I AND2)=2 THENVX=-1
1030  IF (I AND1)=1 THENVX= 1
1040  IF (I AND8)=8 THENVY=-1
1050  IF (I AND4)=4 THENVY= 1
1060  RETURN
1070  '
1100  '●サブルーチン2●
1110  IF X=2 AND Y=4 GOTO1200
```

160行から250行では，風せんの動きとブロックの動きを配列に読み込んでます。配列N(I)はブロック，配列M(I)は風せんの動きです。

300行から450行では，風せんを動かしてます。変数X，Yは風せんの台の位置変数A，Bは風せんの位置です。

470行から490行では，ブロックを動かしてます。

540行のREM文をはずすと，スピードが遅くなります。ゲームが難しいと思う人は，はずしてください。

サブルーチン1は，風せんをコントローラで動かすルーチンです。

サブルーチン2は，ゴールインしたかどうかを調べています。

```
1120 RETURN
1130 '
1200 '●サブ゛ルーチン3●
1210 LOCATE5,10:PRINT"** ゴ゛ール
イン デ゛ス **";
1220 LOCATE5,12:PRINT"ツギ゛ノ メン
ヘ チャレンジ゛!";
1230 S=S+D
1240 LOCATE8,21:PRINT S;
1250 FOR I=1TO5
1260 PLAY"T1B0A0G0F0E0D0C0"
1270 NEXT
1280 END
1290 '
1300 '●ゲ゛ームオーバ゛ー●
1310 LOCATE5,10:PRINT"**ゲ゛ームオ
ーバ゛ー デ゛ス**";
1320 LOCATE5,12:PRINT"モイチド゛ チ
ャレンジ゛ シテネ!";
1330 IF S>H THEN H=S
1340 DEF SPRITE 0,(2,1,0,0,0
)=CHR$(180)+CHR$(181)+CHR$(1
1350 SPRITE 0,A*8+12,B*8+18
1360 FOR I=1TO6
1370 PLAY"T1B0C0B0C0B0C0B0"
1380 NEXT
1390 LOCATE8,21:PRINT S:
1400 INPUT"",I:GOTO120
```

BGグラフィック画面17-A

```
130 PALETB 0,13,48,33,25
140 S=0:D=995:C=0:L=2:R=6
```

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
0	G70	G70	G70	G70	G70	G70	G70	G70	ボ	ー	ナ	ス	=		1	0	0	0	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70
1	G70	G	O	A	L	D40		G70	G70	L	E	V	E	L	=		1	G70	G70	G70						G70	G70	G70
2	G70							G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70									G70	G70
3	G70								G70	G70	G70	G70	G70	G70	G70	G70	G70											G70
4	G70								G70	G70	G70	G70	G70	G70	G70	G70	G70					G70	G70	G70				G70
5	G70		F70						G70	G70	G70	G70	G70	G70							G70	G70	G70	G70				G70
6	G70		F00						G70	G70	G70											G70	G70	G70				G70
7	G70	G70							G70													G70	G70				G70	G70
8	G70	G70	G70	G70													G70	G70									G70	G70
9	G70	G70	G70											G70	G70	G70	G70	G70									G70	G70
10	G70	G70	G70				G70	G70	G70				G70	G70	G70	G70	G70	G70								G70	G70	G70
11	G70	G70				G70	G70	G70	G70					G70	G70	G70	G70	G70				G70	G70	G70	G70	G70	G70	G70
12	G70					G70	G70	G70	G70					G70	G70	G70	G70	G70				G70	G70	G70	G70	G70	G70	G70
13	G70					G70	G70	G70	G70			G70	G70	G70	G70							G70	G70	G70	G70	G70	G70	G70
14	G70					G70	G70	G70				G70	G70	G70	G70								G70	G70	G70	G70	G70	G70
15	G70					G70	G70					G70	G70	G70	G70												G70	G70
16	G70					G70	G70																					G70
17	G70	G70																										G70
18	G70	G70	G70																									G70
19	G70	G70	G70	G70						G70	G70	G70	G70	G70	G70	G70	G70	G70				D40	S	T	A	R	T	G70
20	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70

BGグラフィック画面17-B

```
130 PALETB 0,13,48,33,27
140 S=0:D=995:C=0:L=2:R=6
```

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
0	G70	G70	G70	G70	G70	G70	G70	G70	ボ	ー	ナ	ス	=		1	0	0	0	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70
1	G70	G	O	A	L	D40		G70	G70	L	E	V	E	L	=		3	G70	G70	G70						G70	G70	G70
2	G70							G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70								G70	G70
3	G70							G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70										G70
4	G70							G70	G70				G70	G70	G70	G70	G70	G70					G70	G70				G70
5	G70		F70						G70	7					G70	G70	G70	G70				G70	G70	G70				G70
6	G70		F00														G70	G70			G70	G70	G70	G70				G70
7	G70	G70	G70																		G70	G70	G70	G70				G70
8	G70	G70	G70								G70										G70	G70	G70					G70
9	G70	G70					G70	G70	G70	G70										G70	G70	G70						G70
10	G70					G70	G70	G70						G70	G70	G70	G70	G70	G70	G70	G70	G70						G70
11	G70					G70	G70	G70					G70	G70	G70	G70	G70	G70	G70	G70	G70						G70	G70
12	G70				G70	G70	G70	G70			G70	G70	G70	G70	G70				G70	G70	G70					G70	G70	G70
13	G70	G70				G70	G70				G70	G70	G70	G70										G70	G70	G70	G70	G70
14	G70	G70				G70	G70				G70	G70	G70											G70	G70	G70	G70	G70
15	G70	G70				G70	G70	G70			G70	G70	G70				G70					G70	G70	G70	G70	G70	G70	G70
16	G70	G70	G70			G70											G70	G70	G70									G70
17	G70	G70	G70													G70	G70	G70	G70									G70
18	G70	G70	G70													G70	G70	G70	G70	G70								G70
19	G70	G70	G70						G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70		D40	S	T	A	R	T	G70
20	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70

BGグラフィック画面17-C

```
130 PALETB 0,13,48,33,21
140 S=0:D=995:C=0:L=1:R=9
```

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
0	G70	G70	G70	G70	G70	G70	G70	G70	ボ	ー	ナ	ス	=		1	0	0	0	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70
1	G70	G	O	A	L	D40		G70	G70	L	E	V	E	L	=		3	G70	G70					G70	G70	G70	G70	G70
2	G70						G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70						G70	G70	G70	G70
3	G70						G70										G70	G70	G70							G70	G70	G70
4	G70						G70										G70	G70	G70			G70						G70
5	G70		F70				G70										G70	G70	G70			G70	G70					G70
6	G70						G70				G70	G70	G70	G70	G70		G70	G70	G70			G70	G70	G70				G70
7	G70	G70	G70		G70	G70	G70				G70						G70	G70	G70					G70				G70
8	G70	G70				G70	G70				G70						G70	G70	G70	G70				G70				G70
9	G70	F00					G70				G70						G70	G70	G70					G70	G70			G70
10	G70	G70	G70		G70	G70	G70				G70			G70	G70	G70	G70							G70	G70			G70
11	G70						G70				G70	G70			G70	G70	G70							G70				G70
12	G70						G70				G70	G70			G70	G70				G70	G70	G70	G70	G70				G70
13	G70	G70			G70	G70	G70				G70			G70	G70				G70	G70	G70	G70	G70			G70	G70	G70
14	G70	G70	G70		G70	G70	G70				G70			G70				G70	G70	G70	G70	G70	G70			G70	G70	G70
15	G70	G70	G70			G70	G70				G70						G70	G70	G70	G70	G70	G70	G70	G70		G70	G70	G70
16	G70	G70	G70	G70							G70					G70	G70	G70	G70	G70								G70
17	G70	G70	G70	G70	G70						G70				G70	G70	G70	G70	G70	G70								G70
18	G70	G70	G70	G70	G70	G70					G70			G70	G70	G70	G70	G70	G70	G70								G70
19	G70	G70	G70	G70	G70	G70	G70				G70	G70	G70	G70	G70	G70	G70	G70	G70	G70		D40	S	T	A	R	T	G70
20	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70

BGグラフィック画面17-D

```
130 PALETB 0,13,48,33,20
140 S=0:D=995:C=0:L=8:R=10
```

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
0	M60	M60	M60	M60	M60	M60	M60	M60	ボ	ー	ナ	ス	=		1	0	0	0	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60
1	M60	G	O	A	L	D40		M60	M60	L	F	V	E	L	=		4	M60	M60	M60	M60		M60		M60		M60	M60
2	M60							M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60									M60
3	M60							M60	M60	M60	M60	M60					M60	M60	M60									M60
4	M60							M60	M60	M60	M60	M60					M60	M60	M60					M60				M60
5	M60		F70				M60	M60	M60	M60						M60	M60	M60	M60		M60	M60	M60	M60				M60
6	M60						M60	M60	M60					M60			M60	M60	M60				M60	M60	M60			M60
7	M60						M60	M60					M60	M60			M60	M60	M60				M60	M60			M60	M60
8	M60	M60	M60			M60	M60	M60	M60	M60		M60	M60	M60			M60	M60	M60	M60			M60			M60	M60	M60
9	M60	M60			M60	M60	M60	M60	M60				M60	M60	M60		M60	M60	M60	M60			M60		M60	M60	M60	M60
10	M60	M60		M60	M60	M60	M60	M60	F00					M60	M60			M60	M60				M60			M60	M60	M60
11	M60	M60			M60	M60	M60	M60	M60				M60	M60	M60			M60	M60			M60	M60	M60				M60
12	M60	M60	M60			M60	M60	M60	M60	M60			M60	M60	M60			M60	M60			M60	M60	M60	M60			M60
13	M60	M60	M60	M60		M60	M60	M60	M60	M60	M60		M60	M60	M60		M60	M60	M60				M60	M60	M60			M60
14	M60	M60	M60			M60	M60	M60	M60	M60				M60				M60					M60				M60	M60
15	M60	M60			M60	M60	M60	M60	M60	M60				M60								M60	M60				M60	M60
16	M60	M60		M60	M60	M60		M60	M60	M60	M60		M60	M60						M60	M60	M60						M60
17	M60													M60	M60					M60	M60							M60
18	M60													M60	M60	M60			M60	M60								M60
19	M60	M60					M60						M60	M60	M60	M60	M60	M60	M60	M60		D40	S	T	A	R	T	60
20	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60

BGグラフィック画面17-D

```
130 PALETB 0,13,48,33,17
140 S=0:D=995:C=0:L=13:R=13
```

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
0	G70	G70	G70	G70	G70	G70	G70	G70	ボ	ー	ナ	ス	=		1	0	0	0	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70
1	G70							G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70									G70
2	G70																											G70
3	G70																	F30				F30			F30	F30		G70
4	G70							F30			F30							F30				F30	F30			F30		G70
5	G70		F70					F30	F30		F30	F30			F30	F30		F30				F30	F30			F30		G70
6	G70							F30	F30			F30	F30					F30				F30			F30	F30		G70
7	G70								F30	F30								F30							I	N		G70
8	G70			F30	F30			F30	F30	F30	F30							I	N									G70
9	G70		F30	F30			F30				F30	F30																G70
10	G70		F30	F30		F30	F30				F30	I	N										F30	F30				G70
11	G70		F30	F30			F30	F30						F30	F30		I	N					F30					G70
12	G70			F30	F30			F30					F30	F30				F30				F30	F30					G70
13	G70				F30	F30		F30					F30	F00					F30			F30			F30	F30		G70
14	G70			F30	F30			F30					F30	F30				F30				F30			F30	F30		G70
15	G70			F30			F30	F30						F30	F30			F30				F30	F30		I	N		G70
16	G70		F30	F30		I	N	F30							F30	F30		F30										G70
17	G70		F30																									G70
18	G70		F30																									G70
19	G70																					D40	S	T	A	R	T	G70
20	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70	G70

立体迷路のなかから脱出できるか！

18 立体／三次元迷路

ゲームの遊び方

わたい迷路の中に，まよいこんじゃったんです。

みなさんは，こんな体験あります？　しかも，頼りになるのが，『コンパス』だけ…なんて。ちょっぴり心ぼそいナ。

でも，みなさんと一緒なら，大丈夫だもん！

テレビ画面の真ん中には，迷路の中の様子が表示されてます。迷路の上には，コンパスが，画面の左下には今いる位置があるんです。

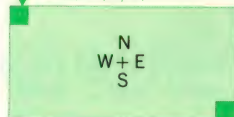
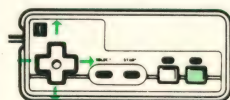
迷路は，左上が(1, 1)，右下が(29, 19)となってます。ゴールは(1, 1)です。迷路は，上が北，下が南，左が西，右が東に向いてます。だから，なるべく北へ，なるべく西へゆくようにするんです。

NORTH＝北
SOUTH＝南
EAST＝東
WEST＝西

ゲームを遊ぶ前に，BGグラフィックを入れて。それからBASICプログラムを入力します。**他の迷路を**試してみたい人は，プログラムの迷路のデータを替えてネ。

このゲームは，立体迷路の中からゴールへ行くんです。最初は，座標（29,19）にいます。ゴールは，座標（1，1）なんです。

コントローラで，見る方角を決めます。方角ですよー，間違えないでネ。一歩進みたい時は，Aボタンを**ポン**って押すの。こうして，迷路を歩いて，ゴールまでゆきますっ！

うーん，でも初めはよく分んないと思いますー。特に，方向感覚が。そのために，方向磁石があるんです。たとえば，コレが『W<NORTH>E』てなってたら，正面が**北**，左側が**西**，右側が**東**ということです。

コントローラを上に押すと北，下に押すと南，左に押すと西，右に押すと東へ向きます。
Aボタンを押すと，進みます。

```
100  '●● GAME/18(チアキ+カナエ) ●●
110  DIM D(1,20),A(15)
120  VIEW:SPRITEOFF
130  PALETB 0,13,48,37,18
150  X=29:Y=19:VX=0:VY=-1
200  RESTORE
210  FORI=0TO20
220  FORJ=0TO1
230  READ I$
240  D(J,I)=VAL("&H"+I$)
250  NEXT
260  NEXT
300  DATA FFFF,FFFE,A088,220A
,BAEB,EEEA,83AB,00AA,FBAB
310  DATA EFAA,82AA,082A,FAAA
,FBAA,8AA2,822A,AABE,BEEA
320  DATA AA80,A28A,BAFF,AEBA
,8008,2882,FFEB,EBFE,882A
330  DATA 2A0A,ABEA,ABEA,A80A
,A822,ABFE,AFAE,A802,8022
340  DATA BFFA,FFEA,8002,000A
,FFFF,FFFE
350  FORI=0TO15
360  READ I$
370  A(I)=VAL("&H"+I$)
380  NEXT
390  DATA 8000,4000,2000,1000
,0800,0400,0200,0100,0080,00
40,0020,0010,0008,0004,0002,
0001
400  '
500  '●メインルーチン●
505  VIEW
510  IFVY=-1  I$="W<NORTH>E"
520  IFVY= 1  I$="E<SOUTH>W"
530  IFVX=-1  I$="S<WEST  >N"
540  IFVX= 1  I$="N<EAST  >S"
550  LOCATE10,3:PRINT I$;
560  LOCATE7,19:PRINT X;Y;"
 ";
610  M=X+VY:N=Y-VX:GOSUB1000
620  IFI=0  M=7:N=5:GOSUB1100
630  M=X-VY:N=Y+VX:GOSUB1000
640  IFI=0  M=20:N=5:GOSUB1100
650  M=X+VX:N=Y+VY:GOSUB1000
660  IFI  Z=6:GOSUB1200:GOTO80
0
670  M=X+VX+VY:N=Y+VY-VX:GOSU
B1000
680  IFI=0  M=9:N=7:GOSUB1100
690  M=X+VX-VY:N=Y+VY+VX:GOSU
B1000
700  IFI=0  M=18:N=7:GOSUB1100
710  M=X+VX*2:N=Y+VY*2:GOSUB1
0
720  IFI  Z=4:GOSUB1200:GOTO80
0
730  M=X+VX*2+VY:N=Y+VY*2-VX:
GOSUB1000
740  IFI=0  M=11:N=9:GOSUB1100
750  M=X+VX*2-VY:N=Y+VY*2+VX:
GOSUB1000
```

（240行）迷路のデータを読み込んでます

（370行）数値のデータを読み込んでます

200行から340行では，迷路の形のデータを，配列D(I, J)に読み込んでます。30行から340行のデータを取り替えると迷路が変わります。

300行から340行のデータ

350行から390行では，迷路のデータを本当の迷路にするための数値を読み込んでます。

500行から560行では，今どの方向を向いてるかを表示してます。それから，今いる位置を表示してます。

610行から780行では，迷路のデータから，目の前に見えるはずの三次元迷路を表示してます。

800行では，ゴールインしたかどうかを調べてます。810行から830行では，コントローラによって方向を変えたり，進んだりします。

サブルーチン１は，迷路の状態を調べるためのルーチンです。変数M，Nに調べたい位置の値を入れてこのルーチンを呼ぶと，変数Iにカベがあるかないかが入るんです。

サブルーチン２では，両側に通り道があるように表示してます。

サブルーチン３では，正面にカベがあるように表示してます。ゴールのカベのときは，ちゃんとゴールのカベを表示します。

サブルーチン４では，一歩進むようにしてます。

サブルーチン５では，コントローラの動きを方向の変数VX，VYに入れてます。

```
760 IFI=0 M=16:N=9:GOSUB1100
770 M=X+VX*3:N=Y+VY*3:GOSUB1000
780 IFI Z=2:GOSUB1200
790 '
800 IFX=1 ANDY=1 GOTO1500
810 I=STICK(0):J=STRIG(0)
820 IFI GOSUB1400:GOTO500
830 IFJ GOSUB1300:GOTO500
840 GOTO800
850 '
1000 '●ｻﾌﾞﾙｰﾁﾝ1●
1010 I=0
1020 J=D(M/16,N)
1030 IF(J AND A(M MOD16))=A(M MOD16) I=1
1040 RETURN
1050 '
1100 '●ｻﾌﾞﾙｰﾁﾝ2●
1110 LOCATEM,N:PRINT"  ";
1120 FORI=N+1TO11
1130 LOCATEM,I:PRINT CHR$(195);
1140 NEXT
1150 RETURN
1160 '
1200 '●ｻﾌﾞﾙｰﾁﾝ3●
1202 I$=CHR$(195)+CHR$(195)
1204 IFM=1 AND N=0 I$=CHR$(210)+CHR$(210)
1210 FORI=12-Z TO11
1220 LOCATE14-Z,I
1230 FORJ=1TOZ
1240 PRINT I$;
1250 NEXT
1260 PRINT
1270 NEXT
1280 RETURN
1290 '
1300 '●ｻﾌﾞﾙｰﾁﾝ4●
1310 M=X+VX:N=Y+VY:GOSUB1000
1320 IFI RETURN
1330 X=X+VX:Y=Y+VY
1340 RETURN
1350 '
1400 'ｻﾌﾞｻｰﾁﾝ5●
1410 VX=(I=2)-(I=1)
1420 VY=(I=8)-(I=4)
1430 RETURN
1440 '
1500 '●ｹﾞｰﾑｵｰﾊﾞｰ●
1510 LOCATE12,8:PRINT"GOAL";
1520 END
```

```
300 DATA FFFF,FFFE,A8A0,0002
,AEAF,EBBA,8888,2AA2,EBFB
310 DATA BABE,A802,8202,AFBA
,BEBA,802A,A28A,FFEA,EFFA
320 DATA 800A,0002,BBFF,EFFE
,AA00,0802,AAFF,FBFA,A888
330 DATA 820A,AFBB,BEEA,8880
,A0AA,EEAE,EFAA,A0AA,0A22
340 DATA BEEB,FABA,8008,0082
,FFFF,FFFE
```

```
300 DATA FFFF,FFFE,A808,2202
,ABAB,AAFE,AA28,AA82,AABE
310 DATA AEBA,8A82,A222,EABA
,BBFA,AAAA,8882,ABAA,EABE
320 DATA 8A22,AAA2,BAFA,AEBA
,A2A2,A202,BEAA,BBBA,82AA
330 DATA 8A0A,EEAE,ABEA,A0A0
,A82A,BEBE,ABAA,80A0,A8AA
340 DATA AFBF,AFBA,A080,2002
,FFFF,FFFE
```

```
300 DATA FFFF,FFFE,A088,0802
,BBAB,FBBE,8A28,8A02,EAFE
310 DATA AEFE,8822,A802,BBAE
,BBFA,82A2,8A0A,FEBA,EAAE
320 DATA A000,8AA2,BFAE,BBBE
,A0AA,A0A2,BAAA,AEBA,8228
330 DATA 0A0A,FAEF,FBEA,8288
,000A,BEBB,FBAA,8882,08A2
340 DATA FFFE,BEBA,8000,808A
,FFFF,FFFE
```

```
300 DATA FFFF,FFFE,8800,2082
,EBFF,FEBE,A880,2022,AEBF
310 DATA AFEA,A2A0,A88A,BAEE
,AABE,802A,AAA2,BEAA,AAAA
320 DATA 888A,AAAA,BBBA,AEBA
,88A0,A882,EEAB,ABBE,88AA
330 DATA 2022,BBAB,BBBA,8A28
,02AA,FFEF,FEAA,8220,A20A
340 DATA BEBF,AAFA,8080,0801
,FFFF,FFFE
```

```
300 DATA FFFF,FFFE,A208,0082
,AAEB,BABE,8288,8A22,FABA
310 DATA EBEA,8A88,2A02,ABEF
,AEBE,8800,A0A2,EFEE,BFAA
320 DATA 8822,A0AA,AFBA,AEAA
,A002,A8AA,BFFF,AFAA,8220
330 DATA A80A,BEBE,AAFE,8000
,1402,BFBF,ABFA,A822,2A2A
340 DATA AAAA,EEAA,8820,2002
,FFFF,FFFE
```

BGグラフィック画面18

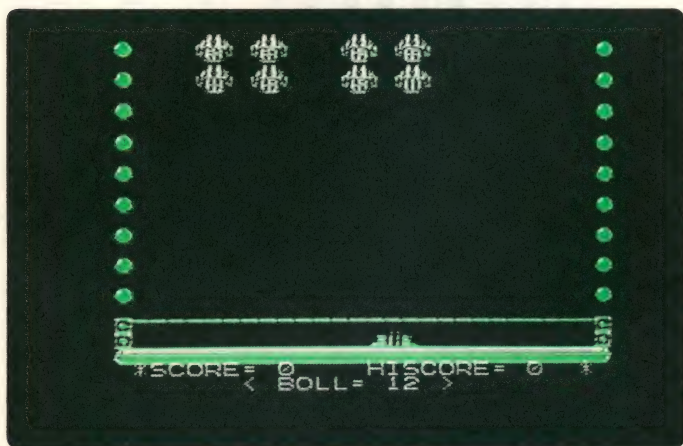
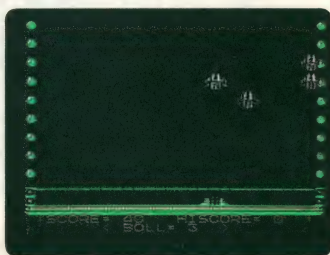
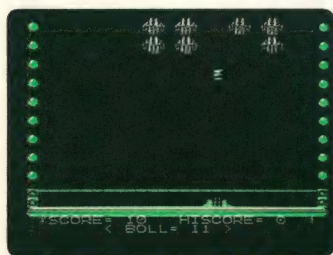
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
0	I60	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	I70
1	J20																											J20
2	J20																											J20
3	J20									G70										G70								J20
3	J20	E40	E40	E40	E40	E40	E40	E40	E40	E40	E40	E40	E40	E40	E40	E40	E40	E40	E40	E40	E40	E40	E40	E40	E40	E40	E40	J20
4	J20	M70	M70	M70	M70	K60		G13													G03		K60	M70	M70	M70	M70	J20
6	J20	M70	M70	M70	M70	K60		G33	G13											G03	G23		K60	M70	M70	M70	M70	J20
7	J20	M70	M70	M70	M70	K60		G33	G33	G13									G03	G23	G23		K60	M70	M70	M70	M70	J20
8	J20	M70	M70	M70	M70	K60		G33	G33	G33	G13							G03	G23	G23	G23		K60	M70	M70	M70	M70	J20
9	J20	M70	M70	M70	M70	K60		G33	G33	G33	G33	G13					G03	G23	G23	G23	G23		K60	M70	M70	M70	M70	J20
10	J20	M70	M70	M70	M70	K60		G33	G33	G33	G33	G33	G13			G03	G23	G23	G23	G23	G23		K60	M70	M70	M70	M70	J20
11	J20	M70	M70	M70	M70	K60		G33	G33	G33	G33	G33	G33	G13	G03	G23	G23	G23	G23	G23	G23		K60	M70	M70	M70	M70	J20
12	J20	M70	M70	M70	M70	K60	I03	I13	I13	I13	I13	I13	I13	I13	I13	I13	I13	I13	I13	I13	I13	I23	K60	M70	M70	M70	M70	J20
13	J20	E50	E50	E50	E50	E50	E50	E50	E50	E50	E50	E50	E50	E50	E50	E50	E50	E50	E50	E50	E50	E50	E50	E50	E50	E50	E50	J20
14	J20																											J20
15	J20																					N	O	R	T	H		J20
16	J20																					○						J20
17	J20																W	E	S	T	○		○	E	A	S	T	J20
18	J20		P	O	S	I	T	I	O	N												○						J20
19	J20		(	X	,	Y	)															S	O	U	T	H		J20
20	J00	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J30	J10

『宇宙人』の侵略を防げ！

19 インベーダー

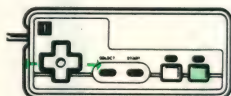
ゲームの遊び方

ドッコドッコと，『インベーダー』が**宇宙から地球へ**，侵入してきました。もう，地球は，タイヘンなんですー。

地球を守る方法は，ただひとつ…。ブロック・ミサイルで，インベーダーをやっつけるだけです。がんばってネ。

テレビ画面の上の方にいるのは，インベーダです。全部で8ヒキでてきます。画面の下の方にいるのは，ミサイル砲です。左右に動かして，ミサイルを発射します。

インベーダが1ヒキでも地上まで来るとゲームオーバーです。

ゲームを遊ぶ前に，BGグラフィックを入れて，それから，BASICのプログラムを入力してネ。

このゲームは，空から降りてくるインベータ群を,**ミサイル砲でやっつけるんです**。インベーダは，8ヒキです。

ミサイルは，12コしかないので，よーく狙って射ってくださいね。全部ミサイルを使ったり，インベーダが地上まで来ると**ゲームオーバー**です。

コントローラで左右に動かします。それから，Aボタンでミサイルを発射します。

```
100  '●●GAME/19(チアキ+カナエ) ●●
110  DIMX(7),Y(7),A(17),B(17)
120  CGSET1,2:VIEW:SPRITE ON
130  PALETB 0,1,48,37,22
150  S=0:L=0
200  LOCATE0,21:PRINT" *SCORE
= 0     HISCORE=";H;"    *"
210  LOCATE7,22:PRINT"< BOLL=
 12  >"
280  I=0
290  FORX=1TO10STEP3
300  FORY=L TO L+2STEP2
310  DEF SPRITE I,(3,1,0,0,0)
="XYZ「"
320  X(I)=X:Y(I)=Y
330  SPRITEI,X*8+12,Y*8+18:I=
I+1
340  NEXT
350  NEXT
360  RESTORE
370  FORI=0TO17
380  READA(I),B(I)
390  NEXT
400  DATA 2,0,2,0,2,0,2,0,2,0
,2,0,2,0,2,0,0,2,-2,0,-2,0,-
2,0,-2,0,-2,0,-2,0,-2,0,-2,0
,0,2
410  A=0:B=12:X=11:Y=20:M=0:M
=0
420  '
500  '●メインルーチン●
510  IFN=1 GOTO1100
520  IFN=0 GOTO1300
550  M=-(M+1)*(M<17)
560  IFA(M)=0 N=-(N=0)
590  IFA=8 GOTO1900
600  GOTO500
610  '
1000  '●サフ゛ーチン1●
1010  IFY<20 RETURN
1020  J=STICK(0)
1030  VX=(J=2)-(J=1):X=X+VX
1050  IF(X=0)+(X=23) X=X-VX
1060  LOCATEX,19:PRINT" ";CHR
$(249);CHR$(226);CHR$(251);"
 ";
1070  IFSTRIG(0)>0 Y=17
1080  RETURN
1090  '
1100  '●サフ゛ルーチン2●
1110  FORI=0TO7
1120  IFX(I)=225 GOTO1180
1130  LOCATEX(I),Y(I):PRINT"
";
1140  X(I)=X(I)+A(M)
1150  Y(I)=Y(I)+B(M)
1160  LOCATEX(I),Y(I):PRINTCH
R$(48+I);
1170  SPRITEI,X(I)*8+12,Y(I)*
8+18
```

```
1180  IFI MOD2=0 GOSUB1000
1190  IFY<20 GOSUB1500
1200  IFY(I)>15 GOTO2000
1210  NEXT
1220  GOTO550
1230  '
1300  '●サブルーチン3●
1310  FORI=7TO0STEP-1
1320  IFX(I)=225 GOTO1380
1330  LOCATEX(I),Y(I):PRINT"
";
1340  X(I)=X(I)+A(M)
1350  Y(I)=Y(I)+B(M)
1360  LOCATEX(I),Y(I):PRINT C
HR$(48+I);
1370  SPRITEI,X(I)*8+12,Y(I)*
8+18
1380  IFI MOD2=0 GOSUB1000
1390  IFY<20 GOSUB1500
1400  IFY(I)>15 GOTO2000
1410  NEXT
1420  GOTO550
1430  '
1500  '●サブルーチン4●
1510  FORJ=1TO2
1520  LOCATEX+2,Y:PRINT" ";
1530  Y=Y-1
1535  P$=SCR$(X+2,Y)
1540  IFP$>" " GOSUB1700:GOSU
B1600:GOTO1570
1550  IFY=0 GOSUB1600:GOTO157
0
1560  LOCATEX+2,Y:PRINTCHR$(1
92);
1570  NEXT
1580  RETURN
1590  '
1600  '●サブルーチン5●
1610  Y=20:J=5:B=B-1
1630  LOCATE14,22:PRINTB;" ";

1640  IFB=0 A=8
```

BGグラフィック画面 19

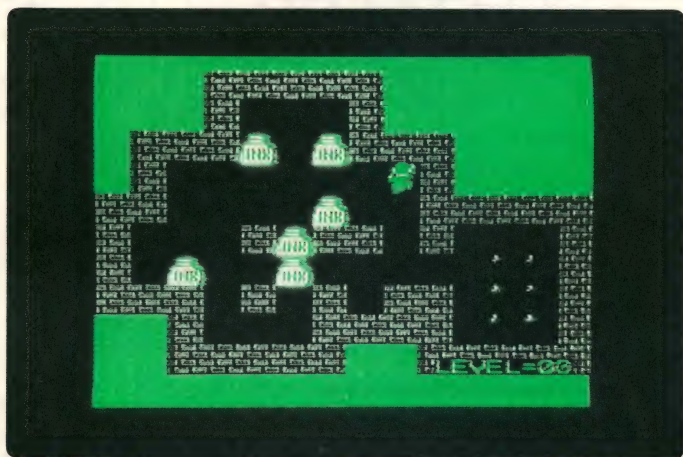
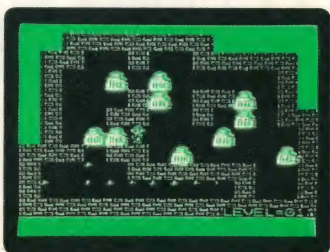
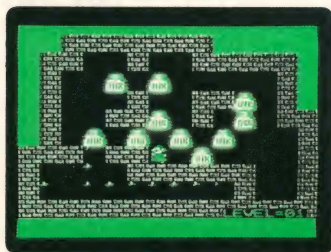
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
0	G72																											G72
1																												
2	G72																											G72
3																												
4	G72																											G72
5																												
6	G72																											G72
7																												
8	G72																											G72
9																												
10	G72																											G72
11																												
12	G72																											G72
13																												
14	G72																											G72
15																												
16	G72																											G72
17																												
18	D44	E00	E00	E00	E00	E00	E00	E00	E00	E00	E00	E00	E00	E00	E00	E00	E00	E00	E00	E00	E00	E00	E00	E00	E00	E00	E00	D44
19	D44												M10	J20	M30													D44
20	F40	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50	F60

倉庫番のアルバイト。いねむりしちゃあ，だめなの

20 倉庫番だって、タイヘンです！

ゲームの遊び方

アルバイト を，してみたいナ…。それで，荷物室の整理をすることになったの。でもネ，荷物室には，『荷物』がバラバラにちらかってるの。もう，いやになっちゃうよぉ…。

うーん，それでも，**一生懸命**，整理整頓しなくっちゃあ！

テレビ画面の真ん中らへんにいるマリオが，あなたです。倉庫の中には，荷物がころがってますね。

ぜんぶ整理整とんできましたー。

次のパターン…。

ゲームを遊ぶ前に，BGグラフィックを入れて。それから，BASICプログラムを入れてネ。でも，パズルのパターンによってBGグラフィックが替ったり，プログラムが変わったりしますー。

このゲームは，食庫の中の **『荷物』を整理整とん** するんです。倉庫の中の〈·〉のマークがあるところに，かたずければいいの。ね。簡単でしょ。終わったら，Bボタン押してね。

自分でオリジナルパターンを作るのが、いいです。

コントローラで動かします。整理整頓ができたら，Bボタンを一。

変数X，Yは，人の位置です。

220から260行では，画面の下の方をぬり絵してるんです。

310行から270行では，コントローラ入力です。390では，普通に動くときそれから400では，荷物を押して動くときです。

サブルーチン1では，荷物を動かせるかなって調べてます。

サブルーチン2では，人を動かします。変数X，Y，VX，VYに値を入れてこのルーチンを呼ぶと，動くんです。

サブルーチン3では，人と荷物を動かします。

```
100  '●● GAME/20(チアキ+カナエ) ●●
120  CGSET1,0:VIEW:SPRITE ON
130  PALETB 2,13,39,16,0
150  X=12:Y=13:VX=-1:VY=0
200  DEF SPRITE 0,(0,1,1,0,0)
=CHR$(0)+CHR$(1)+CHR$(2)+CHR
$(3)
210  SPRITE 0,X*8+16,Y*8+22
220  FOR I=0 TO 27 STEP2
230  COLOR I,21,2
240  LOCATE I,21:PRINT CHR$(2
53);CHR$(253);
250  LOCATE I,22:PRINT CHR$(2
53);CHR$(253);
260  NEXT
270  '
300  '●メインルーチン●
310  I=STICK(0):VX=0:VY=0
320  IF STRIG(0)>0 GOTO1400   ←Bボタンを押したら
330  IF I=0 GOTO300
340  IF I=2 THEN VX=-1 ┐
350  IF I=1 THEN VX= 1 │
360  IF I=8 THEN VY=-1 ├─コントローラ
370  IF I=4 THEN VY= 1 ┘
380  P$=SCR$(X+VX*2,Y+VY*2)
390  IF P$<"(" GOSUB1100
400  IF P$=CHR$(249) GOSUB100
0
410  PLAY"T1B0":PAUSE10
420  GOTO300
430  '
1000  '●サブルーチン1●
1010  P$=SCR$(X+VX*4,Y+VY*4)
1020  IF P$>"'" THEN RETURN
1030  GOSUB1200:GOSUB1200
1040  RETURN
1050  '
1100  '●サブルーチン2●
1110  M=VX+VY+2-((VX*2)<VY)*(
-VX*2-VY*2+4)
1120  DEF MOVE(0)=SPRITE(0,M,
1,8,1,0)
1130  POSITION 0,X*8+16,Y*8+2
2
1140  X=X+VX*2
1150  Y=Y+VY*2
1160  SPRITE 0:MOVE 0
1170  RETURN
1180  '
1200  '●サブルーチン3●
1210  M=VX+VY+2-((VX*2)<VY)*(
-VX*2-VY*2+4)
1220  DEF MOVE(0)=SPRITE(0,M,
1,4,5,0)
1230  POSITION 0,X*8+16,Y*8+2
2
1240  SPRITE 0:MOVE 0
1250  LOCATE X+VX*2,Y+VY*2   :
PRINT"   ";
1260  LOCATE X+VX*2,Y+VY*2+1:
```

```
PRINT"   ";
1270 LOCATE X+VX*3,Y+VY*3  :
PRINT CHR$(249);CHR$(251);
1280 LOCATE X+VX*3,Y+VY*3+1:
PRINT CHR$(250);CHR$(252);
1290 X=X+VX
1300 Y=Y+VY
1310 PAUSE3
1320 RETURN
1330 '
1400 '●サブルーチン4●
1410 RESTORE
1420 READ I,J
1425 IF I+J GOTO1440
1430 IF SCR$(I,J)<>CHR$(249)
 GOTO300
1435 GOTO1420
1440 FOR I=1 TO 5
1450 PLAY"T1C0D0E0F0"
1460 NEXT
1470 LOCATE 5,10:PRINT"** GO
OD!! **";
1480 END
1490 '
1500 '
1510 DATA 22,13,24,13,22,15,
24,15,22,17,24,17,0,0
```

ちゃんと倉庫にしまったか…？

倉庫のデータでは

サブルーチン4では，本当に倉庫にしまったかを調べてます。

BGグラフィック画面20

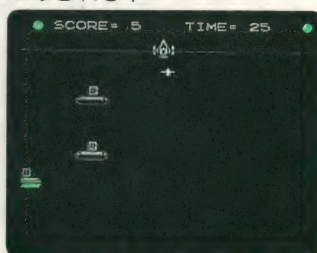
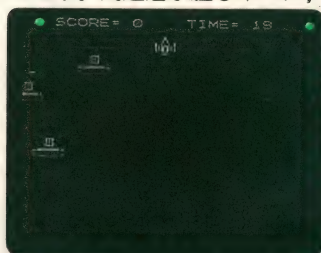
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
0	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50
1	F50	F50	F50	F50	F50	F50	G72	G72	G72	G72	G72	G72	G72	G72	G72	G72	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50
2	F50	F50	F50	F50	F50	F50	G72	G72	G72	G72	G72	G72	G72	G72	G72	G72	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50
3	F50	F50	F50	F50	F50	F50	G72	G72							G72	G72	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50
4	F50	F50	F50	F50	F50	F50	G72	G72							G72	G72	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50
5	F50	F50	G72	G72	G72	G72	G72	G72	M10	M30			M10	M30	G72	G72	G72	G72	G72	G72	F50	F50	F50	F50	F50	F50	F50	F50
6	F50	F50	G72	G72	G72	G72	G72	G72	M20	M40			M20	M40	G72	G72	G72	G72	G72	G72	F50	F50	F50	F50	F50	F50	F50	F50
7	F50	F50	G72	G72															G72	G72	F50	F50	F50	F50	F50	F50	F50	F50
8	F50	F50	G72	G72															G72	G72	F50	F50	F50	F50	F50	F50	F50	F50
9	G72	G72	G72	G72									M10	M30					G72	G72	G72	G72	G72	G72	G72	G72	G72	G72
10	G72	G72	G72	G72									M20	M40					G72	G72	G72	G72	G72	G72	G72	G72	G72	G72
11	G72	G72							G72	G72	M10	M30	G72	G72	G72	G72			G72	G72							G72	G72
12	G72	G72							G72	G72	M20	M40	G72	G72	G72	G72			G72	G72							G72	G72
13	G72	G72			M10	M30					M10	M30											,		,		G72	G72
14	G72	G72			M20	M40					M20	M40															G72	G72
15	G72	G72	G72	G72	G72	G72			G72	G72			G72	G72			G72	G72	G72	G72			,		,		G72	G72
16	G72	G72	G72	G72	G72	G72			G72	G72			G72	G72			G72	G72	G72	G72							G72	G72
17	F50	F50	F50	F50	G72	G72							G72	G72	G72	G72	G72	G72	G72	G72			,		,		G72	G72
18	F50	F50	F50	F50	G72	G72							G72	G72	G72	G72	G72	G72	G72	G72							G72	G72
19	F50	F50	F50	F50	G72	G72	G72	G72	G72	G72	G72	G72	G72	G72	F50	F50	F50	F50	G72	G72	G72	G72	G72	G72	G72	G72	G72	G72
20	F50	F50	F50	F50	G72	G72	G72	G72	G72	G72	G72	G72	G72	G72	F50	F50	F50	F50	G72	L	E	V	E	L	=	0	0	G72

ドイツの誇る潜水艦『Uボート』を，撃破せよ！

21 ディープ・スキャナー

ゲームの遊び方

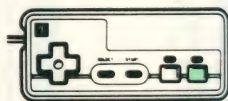
第2次世界大戦の後半のことです。ドイツの誇る潜水艦『Uボート』によって，多くの船がやられていました。

この『Uボート」をやっつけるために，駆逐艦から爆雷を落とすんです。高速潜水艦Uボート，やっつけられる？

海の上にいるのは，駆逐艦です。船には，ちょっと見えないケド，駆逐艦なんですー。
それから，海中には潜水艦がいます。なるべく下の方のをやっつけてネ。

ゲームを遊ぶ前に，**BGグラフィックを**入れて。それから，BASICプログラムを入力してネ。

このゲームは，海の中をとおる『**敵の潜水艦』を爆雷で**やっつけるんです。爆雷は，1度に4コまでポコポコ落とせますー。

ゲームが始まると，テレビ画面の左から潜水艦が出てきます。上手にねらって，爆雷をポコポコ落としてね。ちゃあんと当たると，潜水艦はやられちゃうの。下の方の潜水艦の方が，スコアが高いんですヨ。でね。**タイムが500以上**になっちゃうと，ゲームオーバーです。でも，ゲームオーバーなのに表示されません。そのまま，プログラム終了しちゃいます。

もう1度あそびたいナ，と思ったら…。RUNを実行しないと，だめなの。不便ですねー。**ごめんなさいっ！** でもォ，メモリが少なくって，ちゃんと作れませんでした。

コントローラで駆逐艦は左右に動きます。Aボタンで爆雷を発射するんでーす。

配列Ａ＄(Ｉ)，Ｂ＄(Ｉ)には，潜水艦の形が入ってます。配列Ｘ(Ｉ)には，潜水艦の位置が入ってます。それから，配列Ｍ(Ｉ)には，潜水艦の状態が入ってます

510から550では，潜水艦を動かしてます。560から590では，爆雷を動かしてます。

サブルーチン１では，潜水艦を表示してます。ここを作るのたいへんだったんです。

サブルーチン２では，潜水艦の設定をしてます。

サブルーチン３では，駆逐艦を動かしてます。変数Ｘは，その位置なんです。

```
100  '●● GAME/21(チアキ+カナエ) ●●
110  DIMA$(4),B$(4),X(5),M(5)
,L(3),R(3)
120  CGSET1,1:VIEW:SPRITE ON
130  PALETB 0,1,48,33,17:PALE
TB 1,2,48,41,25:PALETB 2,2,4
8,35,19:PALETB 3,2,48,37,21
200  A$(0)=CHR$(250):A$(1)="
 "+CHR$(250):A$(2)="      "+CH
R$(250):A$(3)="           ":A$(4)="
  ":X=12:C=0
250  B$(0)=CHR$(197)+CHR$(198
):V$(1)=CHR$(236)+CHR$(196)+
B$(0):B$(2)="  "+B$(1):B$(3)
="  "+CHR$(236)+CHR$(196):B$
(4)="   "
300  LOCATE0,4:FORI=1TO99:PRI
NT"     ";:NEXT:FORI=0TO5:M(
I)=-1:NEXT
390  DEF SPRITE 0,(0,1,0,0,0)
="ヒ°フ°ヘ°ホ°":FORI=0TO3:DEF SP
RITE I+1,(3,0,0,0,0)=CHR$(20
9):L(I)=0:NEXT
500  '●メインメーチン●
510  FORI=0TO5
520  IFC MOD(I+1) GOTO550
530  IFM(I)<0 GOSUB1100
540  IFM(I)>-1 GOSUB1000
550  NEXT
560  FORJ=0TO1
570  I=(C MOD2)*2+J:IFL(I) GO
SUB1400
590  NEXT
600  GOSUB1200:C=C+1:IFC>500
END
610  LOCATE20,0:PRINTC;
620  GOTO500
630  '
1000  '●サフ゛ルーチン1●
1010  N=X(I):M=M(I)
1020  LOCATEN,I*3+4:PRINTA$(M
);:LOCATEN,I*3+5:PRINTB$(M);
1040  N=N-(M>1)*2:M=M-(N=0)-(
N>22):X(I)=N:M(I)=M
1070  IFM=5 M(I)=-1
1080  RETURN
1100  '●サフ゛ルーチン2●
1110  IFRND((C/20MOD10)+15) R
ETURN
1120  X(I)=0:M(I)=0:RETURN
1200  '●サフ゛ルーチン3●
1210  J=STICK(0):VX=(J=2)-(J=
1):X=X+VX
1220  IFX=2 OR X=24 X=X-VX
1250  SPRITE0,X*8+16,38
1260  IFSTRIG(0) GOSUB1300
1270  RETURN
1300  '●サフ゛ルーチン4●
1310  I=0
1320  IFL(I)=0 GOTO1350
```

潜水艦のかたちを決めてます（250行を指す）

タイム（610行を指す）

```
1330  I=I+1:IFI=4  RETURN
1340  GOTO1320
1350  L(I)=X+1:R(I)=4:PLAY"T1
C1D1":RETURN
1400  '●サブ゛ルーチン5●
1410  R(I)=R(I)+1
1420  IFR(I)=22 SPRITE I+1:L(
I)=0:RETURN
1440  IFSCR$(L(I),R(I))>" " G
OTO1500
1460  SPRITEI+1,L(I)*8+16,R(I
)*8+22:RETURN
1500  '●サブ゛ルーチン6●
1510  DEF SPRITE 7,(3,1,0,0,0
)=CHR$(176)+"。[]"
1520  SPRITE 7,L(I)*8+12,R(I)
*8+18
1530  FORJ=1TO3:PLAY"T1F0E0D0
C0":NEXT
1540  J=(R(I)-4)/3
1550  LOCATEX(J)+2,J*3+4:PRIN
T"  ";:LOCATEX(J),J*3+5*PRINT
"       ";
1570  M(J)=-1:L(I)=0:SPRITE I
+1
1580  S=S+5+10*J:LOCATE9,0:PR
INTS:
1590  RETURN
```

サブルーチン5では，爆雷を落としてます。

サブルーチン6では，爆雷がどの潜水艦にぶつかったのかを調べてますー。

BGグラフィック画面21

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
0		G73		S	C	O	R	E	=		0					T	I	M	E	=		0					G73	
1																												
2																												
3	E00	E00	E00	E00	E00	E00	E00	E00	E00	E00	E00	E00	E00	E00	E00	E00	E00	E00	E00	E00	E00	E00	E00	E00	E00	E00	E00	E00
4	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50
5	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50
6	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50
7	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50
8	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50	K50
9	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51
10	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51
11	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51
12	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51
13	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52
14	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52
15	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52
16	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52	K52
17	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53
18	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53
19	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53
20	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53	K53

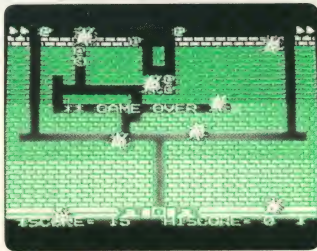
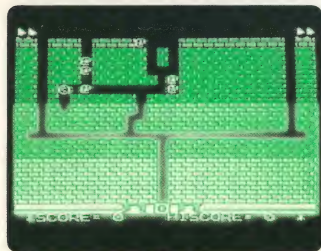
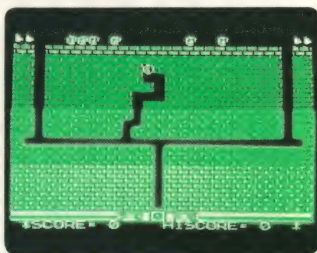
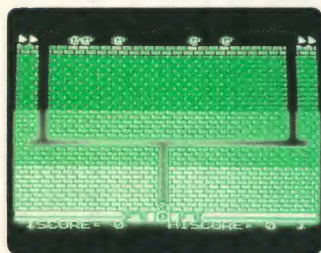
地下基地へ攻めてくる『エイリアン』を，退治します

22 2001年地底の旅

ゲームの遊び方

2001年。地球のエネルギーは，すべて地下に頼っていました。

ところが，このことに気が付いた『エイリアン』が，チャンスとばかりに攻めてきちゃったの。ワッ，どうしましょ！

ンもう，いやな性格のエイリアンさんって，キライなんだからぁ。

テレビ画面の上からエイリアンが落っこちてきます。エイリアンは〈$〉です。画面の下の方には，もぐらタンクがいます〈◯〉。エイリアンは，キゼツすると〈Ⓠ〉になります。

『地面もぐり』をしだしたら，スグに追いかけてネ。基地がやられちゃいます。それから，モグラタンクは画面上下左右に，行き過ぎないようにしないと，だめなのー。エラーが出ちゃいますぅ。

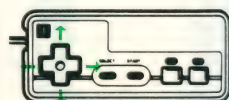
ゲームを遊ぶ前に，BGグラフィックを入れて…。それから，BASICプログラムを入力してネ。

このゲームは，空から落ちてくるエイリアンが，地底の基地に侵入，よいようにガンバルんです！　ねっ。なかなか楽しそうでしょー。もちろん，自信作ですー。

エイリアンは，落ちると少しのあいだ**キゼツしちぁいます**。それに，横に動けないと地面にもぐるんです。この『地面もぐり』は，横に動ける時でも，気が向くとやります。

このエイリアンを退治するには，『もぐらタンクでキゼツしてる時に回収しちゃえば，大丈夫』

でも，エイリアンにぶつかるとゲームオーバーです。

このゲームのコツは，うーん，エイリアンが『地面もぐり』をしないようにする，かなぁ？

『モグラタンク』は，コントロールキーで動かします

配列X(I)，Y(I)はエイリアンの位置の変数です。配列M(I)はエイリアンの状態，V(I)は動きの変数です。

```
100  '●● GAME/22(チアキ+カナエ) ●●
110  DIM X(5),Y(5),M(5),V(5)
120  CGSET1,0:VIEW:SPRITE OFF
130  PALETB 0,13,48,33,17
140  PALETB 1,13,48,35,19
150  PALETB 2,13,48,37,21
160  PALETB 3,13,48,41,25
170  C=0:X=13:Y=19
200  LOCATE0,21:PRINT" *SCORE
= 0     HISCORE=";H;"    *";
210  FORI=0TO5
220  Y(I)=-1:M(I)=0
230  NEXT
240  LOCATEX,Y:PRINTCHR$(180)
;
250  '
400  '●メインルーチン●
410  J=(C MOD3)*2
420  FORI=J TOJ+1
430  IFY(I)=-1 GOSUB1000
440  IFY(I)>-1 GOSUB1100 ← エイリアンを動かすんですー
450  NEXT
500  I=STICK(0)
510  IFI GOSUB1500 ← もぐらタンクを動かします
540  C=C+1
550  GOTO400
560  '
```

サブルーチン1では，エイリアンの位置を決めてます。

```
1000  '●サフ゛ルーチン1●
1010  X(I)=RND(20)+3
1020  Y(I)=0
1030  PLAY"T1B1C1B1C1"
1040  LOCATEX(I),Y(I):PRINT"$
";
1050  RETURN
1060  '
```

サブルーチン2では，エイリアンの動きを決めてます。落ちるならサブルーチン3を，横に動くならサブルーチン4を呼びます。

```
1100  '●サフ゛ルーチン2●
1110  P$=SCR$(X(I),Y(I)+1)
1120  IFP$=" " GOSUB1300:RETU
RN
1130  IFP$>" " GOSUB1400:RETU
RN
1140  IFP$=CHR$(180) GOTO2000
1150  RETURN
1160  '
```

サブルーチン3では，エイリアンが落ちるようにしてます。落っこちた後，配列M(I)にキゼツする時間の値を入れてます。

```
1300  '●サフ゛ルーチン3●
1310  LOCATEX(I),Y(I):PRINT"
";
1320  Y(I)=Y(I)+1
1330  LOCATEX(I),Y(I):PRINT"@
";
1335  P$=SCR$(X(I),Y(I)+1)
1340  IFP$=" " GOTO1300
1350  IFP$=CHR$(180) GOTO2000
1360  PLAY"T1E1F1E1F1E1F1E1"
1370  M(I)=10-S/30 ← 少しの間キゼツ！
1380  RETURN
1390  '
```

サブルーチン4では，エイリアンを横に動かしています。動くとこに，何かあったらサブルーチン7を呼びます。

```
1400  '●サフ゛ルーチン4●
1410  M(I)=M(I)+(M(I)>0)
1420  IFM(I) RETURN
1430  IFV(I)=0 V(I)=RND(7)-3
1440  LOCATEX(I),Y(I):PRINT"
";
```

```
1450  IFSCR$(X(I)+SGN(V(I)),Y
(I))>" " GOSUB1800
1460  X(I)=X(I)+SGN(V(I))
1470  LOCATEX(I),Y(I):PRINT"$
";
1475  IFY(I)>19 GOTO2000
1480  V(I)=V(I)-SGN(V(I))
1490  RETURN
1495  '
1500  '●サブルーチン5●
1510  VX=(I=2)-(I=1)  ┐
1520  VY=(I=8)-(I=4)  ┘── キー入力
1530  P$=SCR$(X+VX,Y+VY)
1540  IFP$="$" GOTO2000
1550  IFP$="@" GOSUB1700
1560  LOCATEX,Y:PRINT" ";
1570  X=X+VX
1580  Y=Y+VY
1590  LOCATEX,Y:PRINTCHR$(180
);
1600  RETURN
1610  '
1700  '●サブルーチン6●
1710  PLAY"T1C2D2"
1720  FORI=0TO5
1730  IFX(I)=X+VX AND Y(I)=Y+
VY Y(I)=-1
1740  NEXT
1750  S=S+5 ←── スコア加算
1760  LOCATE8,21:PRINTS;
1770  IFS=250 GOTO2200
1780  RETURN
1790  '
1800  '●サブルーチン7●
1810  IFP$=CHR$(180) GOTO2000
1820  IFSCR$(X(I)-SGN(V(I)),Y
(I))>" " Y(I)=Y(I)+1
1840  V(I)=0
1850  IF RND(600-C)<3 Y(I)=Y(
I)+1
1860  RETURN
1870  '
2000  '●ゲームオーバー●
2010  SPRITE ON
2020  FORI=0TO7
2030  DEF SPRITE I,(3,1,0,0,0
)=CHR$(180)+CHR$(181)+CHR$(1
82)+CHR$(183)
2040  SPRITE I,RND(28)*8+16,R
ND(21)*8+22
2050  NEXT
2060  FORI=1TO5
2070  PLAY"T1B0A0G0F0E0D0C0"
2080  NEXT
2090  LOCATE5,9:PRINT"** GAME
 OVER **";
2100  INPUT"",I:GOTO120
2110  '
2200  '●ゲームオーバー●
2210  LOCATE5,9:PRINT"** YOU
WIN!! **";
2220  INPUT"",I:GOTO120
```

サブルーチン5では，もぐらタンクを動かしてます。

サブルーチン6では，エイリアンの回収をしてます。1つ回収するとスコアに5点プラスします。

サブルーチン7では，エイリアンがぶつかった時の処理をしてます

BGグラフィック画面22

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
0																												
1	F73	F73																									F73	F73
2	F33	F33		F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33	F33		F33	F33
3	F32	F32		F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32		F32	F32
4	F32	F32		F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32		F32	F32
5	F32	F32		F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32		F32	F32
6	F32	F32		F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32		F32	F32
7	F32	F32		F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32		F32	F32
8	F32	F32		F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32		F32	F32
9	F31	F31		F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31		F31	F31
10	F31	F31		F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31		F31	F31
11	F31	F31		F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31		F31	F31
12	F31																											F31
13	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31		F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31
14	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31		F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31
15	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30		F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30
16	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30		F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30
17	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30		F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30
18	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30		F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30
19	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30		F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30	F30
20	F50	F50	F50	F50	F50	F50	F50	F50	F50	F60	M10	M30	D70	D40	D50	M10	M30	F40	F50	F50	F50	F50	F50	F50	F50	F50	F50	F50

ふたりで，リンゴ狩りをします

23 お花畑で、いっしょにネ！

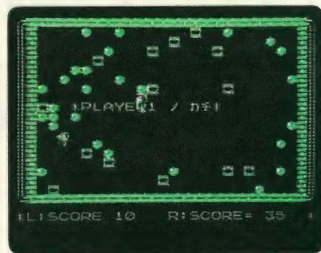
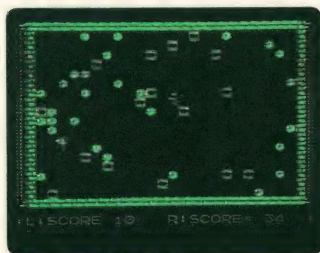
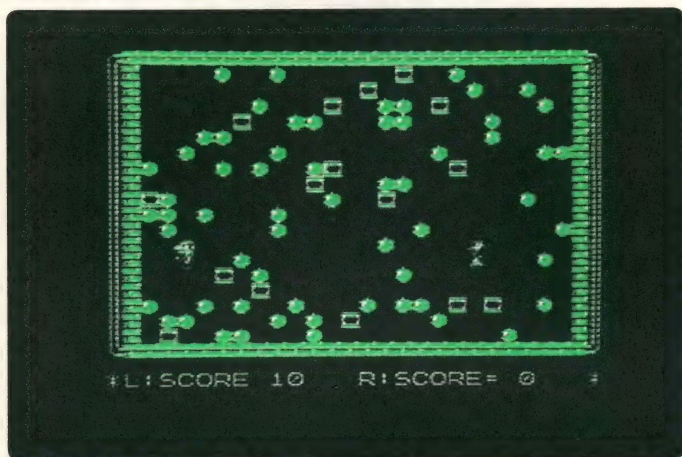
ゲームの遊び方

ふたりで，**リンゴ狩り**をするんです…。

でもぉ，リンゴの数は，決まってます。そうすると，どーしても競争になっちゃいますよねー。ほんと，こまっちゃうの。

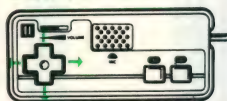
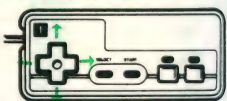
ケンカしないように，リンゴ狩りをたのしみましょう。

テレビ画面のまわりにあるのは，カベです。画面中には，レンガとリンゴがたくさんあります。
コントローラＩの人は，マリオ，コントローラⅡの人はレディを動かします。

カベとかブロックにぶつからないようにしてね。ぶつかっちゃうとゲームオーバーですー！

ゲームを遊ぶ前に，ＢＧグラフィックを入れて，それから，ＢＡＳＩＣプログラムを入力してネ。

このゲームは、**2人で遊ぶ**んです。なので，「ひとりっこ」の人はお友だちと遊んでね。これは，2人で画面にちらばった『リンゴ』を取るの。**たくさん取った方の勝ちー。**

コントローラー Ｉの人は『マリオ』コントローラⅡの人は『レディ』を動かします。カベやブロックにぶつからないでネ。

●コルトローラＩ

●コントローラⅡ

『マリオ』は，コントローラＩで動かします。『レディ』は，コントローラⅡで動かします。

配列X(I)，Y(I)は，それぞれマリオ，レディの位置です。
210から260行では，ブロックを表示してます。270から320行では，リンゴを表示してます。

410から420行では，マリオを，430から440行ではレディを動かしてます。

サブルーチン1では，マリオ，レディを動かしてます。

サブルーチン2では，どっちが勝ったのかを調べてます。

```
100  '●● GAME/23(チアキ+カナエ) ●●
110  DIM X(1),Y(1),S(1)
120  CGSET1,0:VIEW:SPRITE ON
130  PALETB 0,2,48,40,22
140  PALETB 1,2,48,39,21
150  X(0)=5:Y(0)=13
160  X(1)=15:Y(1)=13
170  PALETS 0,2,37,20,26
200  LOCATE 0,21:PRINT"*L:SCO
RE= 0   R:SCORE= 0   *";
210  FOR I=0 TO 15
220  X=RND(26)+1
230  Y=RND(18)+1
240  IF SCR$(X,Y)>" " GOTO220
250  LOCATE X,Y:PRINT CHR$(20
9);
260  NEXT
270  FOR I=0 TO 61
280  X=RND(26)+1
290  Y=RND(18)+1
300  IF SCR$(X,Y)>" " GOTO280
310  LOCATE X,Y:PRINT CHR$(21
5);
320  NEXT
330  LOCATE X(0),Y(0):PRINT"
";
340  LOCATE X(1),Y(1):PRINT"
";
350  '
400  '●メインルーチン●
410  J=STICK(0):I=0  ←マリオの場合です
420  IF J GOSUB1000
430  J=STICK(1):I=1  ←レディの場合です
440  IF J GOSUB1000
450  GOTO400
460  '
1000  '●サブルーチン1●
1010  VX=(J=2)-(J=1)
1020  VY=(J=8)-(J=4)
1030  P$=SCR$(X(I)+VX,Y(I)+VY
)
1040  IF P$>" " GOSUB1200
1050  M=VX+VY+2-((VX+VX)<VY)*
(-VX-VX-VY-VY+4)
1060  DEF MOVE(1)=SPRITE(I,M,
1,4,0,0)
1070  POSITION I,X(I)*8+12,Y(
I)*8+16
1080  MOVE I  ←動かしてます
1090  X(I)=X(I)+VX
1100  Y(I)=Y(I)+VY
1110  RETURN
1120  '
1200  '●サブルーチン2●
1210  IFP$=CHR$(215) GOTO1250
1220  IFP$=CHR$(209) GOTO1450
1230  IFP$=CHR$(195) GOTO1450
1240  '
1250  S(0)=S(0)+1:S=S+1
1260  IF S=60 GOTO1400
```

```
1270 LOCATE I+I*14,21:PRINT
S(I);
1280 LOCATE X(I)+VX,Y(I)+VY:
PRINT"  ";
1290 PLAY"T1C0B0"
1300 RETURN
1310 '
1400 '●サブルーチン3●
1410 IF S(0)>S(1) GOTO1500   ← マリオの勝ち
1420 IF S(1)>S(0) GOTO1600   ← レディの勝ち
1430 END
1440 '
1450 IF I=1 GOTO1500   ← マリオの勝ち
1460 IF I=0 GOTO1600   ← レディの勝ち
1470 END
1480 '
1500 '●ゲームオーバー1●
1510 LOCATE 5,9:PRINT"*PLAYE
R1 ノ カチ*";
1520 INPUT"",I:GOTO120
1530 '
1600 '●ゲームオーバー2●
1610 LOCATE 5,9:PRINT"*PLAYE
R2 ノ カチ*";
1620 INPUT"",I:GOTO120
```

サブルーチン3では，どっちが勝ったのかを調べてます。

BGグラフィック画面23

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
0	I61	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	I71
1	J21	F31																									F31	J21
2	J21	F31																									F31	J21
3	J21	F31																									F31	J21
4	J21	F31																									F31	J21
5	J21	F31																									F31	J21
6	J21	F31																									F31	J21
7	J21	F31																									F31	J21
8	J21	F31																									F31	J21
9	J21	F31																									F31	J21
10	J21	F31																									F31	J21
11	J21	F31																									F31	J21
12	J21	F31																									F31	J21
13	J21	F31																									F31	J21
14	J21	F31																									F31	J21
15	J21	F31																									F31	J21
16	J21	F31																									F31	J21
17	J21	F31																									F31	J21
18	J21	F31																									F31	J21
19	J01	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	F31	J11
20																												

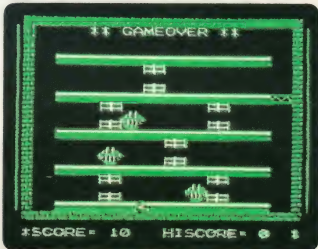
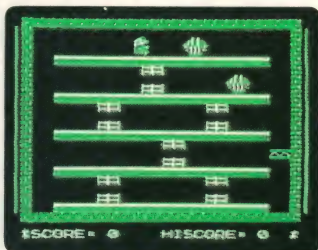
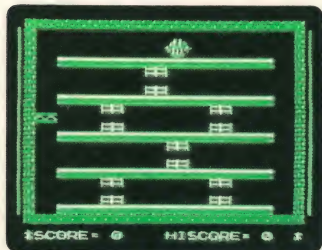
エレベーターを上手に利用して，エイリアンをやっつけてッ！

24 恐怖のエレベーター・エイリアン

ゲームの遊び方

ビルのなかに『エイリアンさん』が，入ってきちゃったの。

このエイリアンさん，とっても強くって…。だって，歩くのはやいし，ひとだってパクパクたべちゃうし。こわーい。

でもね，ひとつだけ，弱点があるから，だいじょうぶ。

テレビ画面には，5階だてのビルがあります。それぞれの階には，ハシゴがかかってます。
それから，エイリアンなんかも歩いてますネー。5階にいるマリオを動かすんですよ。

エレベータに乗るのって，とてもタイヘンです。なれるまで練習してね。

このゲームを遊ぶ前に，BGグラフィックを入れて…。それから，BASICプログラムを入力してネ。**BGグラフィックは，**みなさんが好きなのを選べばいいの。

で，**RUNします。** すると，テレビ画面にはエイリアンがペッコペコ，つぎつぎに歩いてきます。でも，『マリオ』がいないんですー。どうしたの?。うーん，じつはコントローラを押すまで，出てこないんですネ。

このゲームは，ハシゴの上からエイリアンにのっかって，やっつけちゃうんです。エレベータに乗って，上へあがることもできるんですよー。えと，**エレベータの乗り方が，**ちょっぴりむつかしいの。床と同じ高さになったらタイミングよく乗って，そのあとコントローラを上に押しててね。

このゲームは，ぜひ遊んでもらいたあいの。ウン！

コントローラを左右に動かすと，マリオが動きます。エレベータに乗ったときは，コントローラを上に押さないと，落っこちちゃうんです。AボタンかBボタンを押すと，ハシゴを降りれます。エイリアンの上に落ちれば，いいのー。

配列D(I)は，エレベータの動きの変数です。配列V(I)とY(I)とVY(I)には，エイリアンの位置とか動く方向が入ってます。160行では，マリオの形を決めてます。220行では，エレベータの動きを決めてます。250行では，エイリアンの形を決めてます。

410から440行でエイリアンを動かしてます。460から470行で，エレベーターを動かしてます。500から520行で，マリオを動かしてます。

サブルーチン1では，エイリアンを出す準備をしてます。

サブルーチン2では，エイリアンを動かしてます。マリオを食べると1800行へジャンプしますー。

サブルーチン4では，エレベータを動かしてます

```
100  '●● GAME24(チアキ+カナエ) ●●
110  DIM D(9),X(4),Y(4),VX(4)
,A$(2)
120  CGSET1,0:VIEW:SPRITE ON
130  PALETB 2,13,44,37,21
140  PALETB 0,13,48,39,22
150  S=0:C=0:B=2:A=9:X=12:Y=2
:VX=1
160  A$(0)=CHR$(0)+CHR$(1)+CH
R$(2)+CHR$(3):A$(1)=CHR$(20)
+CHR$(21)+CHR$(22)+CHR$(23):
A$(2)=CHR$(1)+CHR$(0)+CHR$(3
)+CHR$(2)
200  FORI=1TO26:COLOR I,21,2:
LOCATE I,21:PRINT CHR$(195);
:NEXT
210  LOCATE 0,23:PRINT" *SCOR
E= 0      HISCORE=";H;"   *";
220  FORI=0TO9:D(I)=20-I*2:NE
XT
250  FORI=0TO4:VX(I)=0:DEF SP
RITE I,(0,1,0,0,0)="XYZ「":SP
RITE I:NEXT:SPRITE 7
400  '●メインルーチン●
410  FORI=0TO4
420  IFVX(I)=0 GOSUB1000   ←エイリアンを表示しない
430  IFVX(I) GOSUB1100     ←エイリアンを表示
440  NEXT
460  C=C+1:IFC>300 GOTO1800
470  IFC MOD2 GOSUB1300
500  I=STICK(0)
510  IFI GOSUB1400
520  IFSTRIG(0) GOSUB1600
550  GOTO400
1000  '●サフ゛ルーチン1●
1010  IFRND(30-C/10)>0 RETURN

1020  J=RND(2)
1030  IFJ=0 X(I)=2:Y(I)=2+RND
(5)*4:VX(I)=11
1040  IFJ=1 X(I)=24:Y(I)=2+RN
D(5)*4:VX(I)=-11
1050  PLAY"T1D0C0"   ←ピポ!と音!
1060  RETURN
1100  '●サフ゛ルーチン2●
1110  IFABS(VX(I))=1 SPRITE I
:VX(I)=0:RETURN
1120  X(I)=X(I)+SGN(VX(I))*2
1130  P$=SCR$(X(I),Y(I))+SCR$
(X(I)+1,Y(I))
1140  SPRITE I,X(I)*8+16,Y(I)
*8+22
1150  IFP$<>"  " GOTO1800
1160  VX(I)=VX(I)-SGN(VX(I))
1180  RETURN
1300  '●サフ゛ルーチン3●
1310  LOCATEB,D(A):PRINT"  ";
1320  A=C/2 MOD10:IFA=9 B=2-(
B=2)*22
1330  LOCATEB,D(A):PRINTCHR$(
```

```
208);CHR$(208);
1340  RETURN
1400  '●サブルーチン4●
1410  VX=(I=2)-(I=1)
1420  LOCATEX,Y:PRINT"  ";
1430  DEF SPRITE 7,(0,1,0,-(V
X=1),0)=A$(VX+1):X=X+VX
1440  SPRITE 7,X*8+16,Y*8+22
1450  LOCATEX,Y:PRINT".";
1460  P$=SCR$(X-(VX=1),Y+2):I
FP$<>CHR$(197) GOSUB1500
1480  RETURN
1500  '●サブルーチン5●
1510  IFP$<>CHR$(208) GOTO180
0
1520  IFC MOD2=0 Y=Y-2 ←エレベータとともにのぼる
1530  RETURN
1600  '●サブルーチン6●
1610  P$=SCR$(X,Y+3)+SCR$(X+1
,Y+3):IFP$<>CHR$(210)+CHR$(2
10) RETURN
1620  LOCATEX,Y:PRINT"  ";
1630  FORI=1TO4:Y=Y+1
1640  SPRITE7,X*8+16,Y*8+22:P
LAY"T1B0A0G0F0"
1650  NEXT:LOCATEX,Y:PRINT"."
;
1660  FORI=0TO4
1670  IFX=X(I)ANDY=Y(I) GOSUB
1700
1680  NEXT
1690  RETURN
1700  '●サブルーチン7●
1710  S=S+10:LOCATE8,23:PRINT
S;
1720  SPRITE I:VX(I)=0 ←エイリアンを消します
1730  RETURN
1800  '●サブルーチン8●
1810  FORI=Y TO20:SPRITE 7,X*
8+16,I*8+22:PAUSE10:NEXT
1820  PLAY"T1B5A5G5F5"
1830  LOCATE7,1:PRINT"** GAME
OVER **";
1840  IFS>H H=S
1850  INPUT"",I:GOTO120
```

サブルーチン4では，マリオを動かしています。変数X，Yは，その位置が入ってます。マリオの下が床じゃなかったらサブルーチン5を呼びます。

サブルーチン5では，エレベータに乗ったのか，床から落ちちゃったのかを調べてます。

サブルーチン6では，ハシゴをおりてます。ハシゴの下にエイリアンがいたらサブルーチン7を呼びます。

サブルーチン7では，スコアに10点プラスしてます。

BGグラフィック画面24

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
0		F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	
1	K62	F32			G70																			G70			F32	K62
2	K62	F32																									F32	K62
3	K62	F32																									F32	K62
4	K62	F32			F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52			F32	K62
5	K62	F32			G70						H22	H22					H22	H22						G70			F32	K62
6	K62	F32																									F32	K62
7	K62	F32									H22	H22					H22	H22									F32	K62
8	K62	F32			F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52			F32	K62
9	K62	F32			G70		H22	H22													H22	H22		G70			F32	K62
10	K62	F32																									F32	K62
11	K62	F32					H22	H22													H22	H22					F32	K62
12	K62	F32			F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52			F32	K62
13	K62	F32			G70						H22	H22					H22	H22						G70			F32	K62
14	K62	F32																									F32	K62
15	K62	F32									H22	H22					H22	H22									F32	K62
16	K62	F32			F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52			F32	K62
17	K62	F32			G70		H22	H22													H22	H22		G70			F32	K62
18	K62	F32																									F32	K62
19	K62	F32					H22	H22													H22	H22					F32	K62
20	K62	F32			F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52			F32	K62

BGグラフィック画面24-B

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
0		F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	
1	K62	F32			G70																			G70			F32	K62
2	K62	F32																									F32	K62
3	K62	F32																									F32	K62
4	K62	F32			F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52			F32	K62
5	K62	F32			G70		H22	H22			H22	H22					H22	H22			H22	H22		G70			F32	K62
6	K62	F32																									F32	K62
7	K62	F32					H22	H22			H22	H22					H22	H22			H22	H22					F32	K62
8	K62	F32			F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52			F32	K62
9	K62	F32			G70		H22	H22			H22	H22					H22	H22			H22	H22		G70			F32	K62
10	K62	F32																									F32	K62
11	K62	F32					H22	H22			H22	H22					H22	H22			H22	H22					F32	K62
12	K62	F32			F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52			F32	K62
13	K62	F32			G70		H22	H22			H22	H22					H22	H22			H22	H22		G70			F32	K62
14	K62	F32																									F32	K62
15	K62	F32					H22	H22			H22	H22					H22	H22			H22	H22					F32	K62
16	K62	F32			F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52			F32	K62
17	K62	F32			G70		H22	H22			H22	H22					H22	H22			H22	H22		G70			F32	K62
18	K62	F32																									F32	K62
19	K62	F32					H22	H22			H22	H22					H22	H22			H22	H22					F32	K62
20	K62	F32			F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52	F52			F32	K62

2ヒキの『ガラゴロン』を，ブロックで囲んでネ

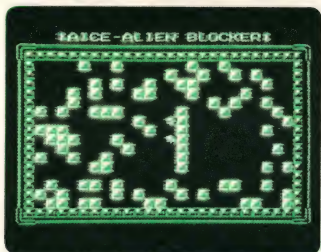
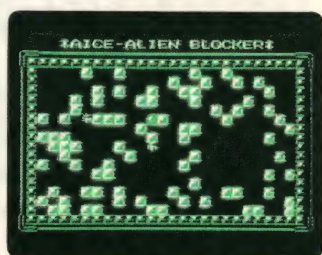
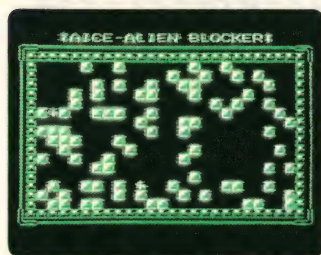
25 ヒョロロンとガラゴロン

ゲームの遊び方

昔むかし…。あるところに，『ガラゴロン』が2ヒキと『ヒョロロン』がいました。ガラゴロンは，ヒョロロンが大好物で，なにかと追いかけてきます。ヒョロロンだって，食べられたくないもん。

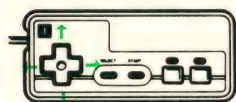
どうにか，**ブロック**で囲って助かりたいんです。

テレビ画面には，たっくさんブロックが表示されます。それから，エイリアンさんやブロックマンなんかも登場します。

まずワナを作っておいて，おびきよせますー。で，後からだんだん小さくするんです。

このゲームを遊ぶ前に，BGグラフィックを入れて…。それから，**BASICプログラム**を入力しておいてネ。

RUNすると，テレビ画面にはブロックがたくさん，『エイリアン』が2ヒキ，『ブロックマン』がでてきますー。

このゲームは，ブロックでエイリアンをとじこめちゃうんです。とにかく，とじこめちゃえば，勝ち！
でも，エイリアンに食べられちゃうと，ゲームオーバーなんです。うーんと，このゲームのコツは，まず，ワナを仕掛けるの。ブロックで，**大きめの囲いを作っておいて**，だんだん小さくしてっちゃいます。

あとは，そーですね，ブロックで壁を作っておくことくらいかナ？こうして，2ヒキのエイリアンをとじこめちゃいますー。でもぉ，**なかなか上手にできません**。やだなー。

コントローラで，上下左右に動かせます

```
100  '●● GAME/25(チアキ+カナエ) ●●
110  DIM X(1),Y(1),A(7),B(7)
120  VIEW:SPRITE OFF
140  PALETB 1,13,48,21,4
150  O=-1:S=0:X=13:Y=10
210  FOR I=1 TO 90
220  LOCATE RND(24)+2,RND(16)
+3:PRINT CHR$(192);
230  NEXT
240  A(0)=0:B(0)=-1:A(1)=1:B(
1)=-1:A(2)=1:B(2)=0:A(3)=1:B
(3)=1:A(4)=0:B(4)=1:A(5)=-1:
B(5)=1:A(6)=-1:B(6)=0:A(7)=-
1:B(7)=-1
250  FOR I=0 TO 1
260  X(I)=RND(24)+2
270  Y(I)=RND(16)+3
280  LOCATE X(I),Y(I):PRINT"$
";  ←――――エイリアンさんを表示
290  PLAY"T1B5A5":PAUSE50
300  NEXT
310  LOCATE X,Y:PRINT CHR$(20
7);
320  '
400  '●メインルーチン●
410  FOR C=0 TO 1
420  IF X(C) GOSUB1000
450  I=STICK(0):IFI=0 GOTO540
460  VX=(I=2)-(I=1)
470  VY=(I=8)-(I=4)
480  P$=SCR$(X+VX,Y+VY)
490  IF P$<>" " GOSUB1300
500  LOCATE X,Y:PRINT" ";
510  X=X+VX
520  Y=Y+VY
530  LOCATE X,Y:PRINT CHR$(20
7);
540  NEXT:IF O GOTO400
550  GOTO1400
560  '
1000  '●サブルーチン1●
1010  I=RND(2)
1020  IF I GOSUB1100←――ランダムに動きます
1030  IF I=0 GOSUB1200←――追っかけます
1040  IF P$=CHR$(207) O=0
1050  LOCATE X(C),Y(C):PRINT"
 ";
1060  X(C)=X(C)+VX
1070  Y(C)=Y(C)+VY
1080  LOCATE X(C),Y(C):PRINT"
$";
1090  RETURN
1095  '
1100  '●サブルーチン2●
1110  J=0
1120  I=RND(8)
1130  P$=SCR$(X(C)+A(I),Y(C)+
B(I))
1140  IF P$<>CHR$(192) VX=A(I
):VY=B(I):RETURN
```

配列X(I)，Y(I)には，エイリアンの位置が入ってます。配列A(I)，B(I)にはエイリアンが周りを調べるときのデータが入ってます。

420行では，エイリアンが表示されるならサブルーチン1を呼んでます。

450から530行では，人を動かしてます。

サブルーチン1では，エイリアンを動かしてます。

サブルーチン2では，ランダムな動きにします。

サブルーチン3では，追っかける動きにします。

サブルーチン4では，ブロックを押してます。

```
1150  I=-(I+1)*(I<7):J=J+1
1160  IF  J<8  GOTO113Ø
1170  X(C)=Ø:S=S+1:IF  S<2  RET
URN
118Ø  RETURN
119Ø  '
12ØØ  '●サフ゛ルーチン3●
121Ø  VX=SGN(X-X(C))
122Ø  VY=SGN(Y-Y(C))
123Ø  P$=SCR$(X(C)+VX,Y(C)+VY
)
124Ø  IF  P$<>CHR$(192)  RETURN
125Ø  GOTO11ØØ
126Ø  '
13ØØ  '●サフ゛ルーチン4●
131Ø  IF  P$="$"  O=Ø
132Ø  PLAY"T1CØDØEØFØ"
133Ø  IF  SCR$(X+VX*2,Y+VY*2)<
>"  "  VX=Ø:VY=Ø:RETURN
134Ø  LOCATE  X+VX*2,Y+VY*2:PR
INT CHR$(192);
135Ø  RETURN
136Ø  '
14ØØ  '●ケ゛ームオーハ゛ー●
143Ø  PLAY"T1B3A3G3F3E3D3C3"
144Ø  LOCATE  7,1Ø:PRINT"*GAME
 OVER*";
145Ø  LOCATE  7,12:PRINT"モウ  サ゛
ンネンテ゛シタ!";
146Ø  INPUT"",I:GOTO12Ø
147Ø  '
15ØØ  '●ケ゛ームオーハ゛ー●
151Ø  LOCATE  7,1Ø:PRINT"*  YOU
 WIN  *";
152Ø  INPUT"",I:GOTO12Ø
```

（1210–1220行）追っかけます

BGグラフィック画面25

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
0					*	A	I	C	E	—	A	L	I	E	N		B	L	O	C	K	E	R	*				
1	I61	J31	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	J31	I71
2	J21	F01	F01	F01	F01	F01	F01	F01	F01	F01	F01	F01	F01	F01	F01	F01	F01	F01	F01	F01	F01	F01	F01	F01	F01	F01	F01	J21
3	K61	F01																									F01	K61
4	K61	F01																									F01	K61
5	K61	F01																									F01	K61
6	K61	F01																									F01	K61
7	K61	F01																									F01	K61
8	K61	F01																									F01	K61
9	K61	F01																									F01	K61
10	K61	F01																									F01	K61
11	K61	F01																									F01	K61
12	K61	F01																									F01	K61
13	K61	F01																									F01	K61
14	K61	F01																									F01	K61
15	K61	F01																									F01	K61
16	K61	F01																									F01	K61
17	K61	F01																									F01	K61
18	K61	F01																									F01	K61
19	J21	F01	F01	F01	F01	F01	F01	F01	F01	F01	F01	F01	F01	F01	F01	F01	F01	F01	F01	F01	F01	F01	F01	F01	F01	F01	F01	J21
20	J01	J31	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	K51	J31	J11

一生懸命 はよして
ちょっとひとやすみ

ゲームを遊ぶのは、ひとまず『おやすみ』にして…。ここでちょっと、頭の体操！

ここでは，ゲームはおやすみして，カンタンなクイズを出してみたいと思います。

でも，カンタンといっても，BASICプログラムをちゃーんと作れないと，だめですョ。すぐには，解けないかもしれないけど，じっくり，ゆっくり考えてネ。

それじゃあ，第1問…。

第1問

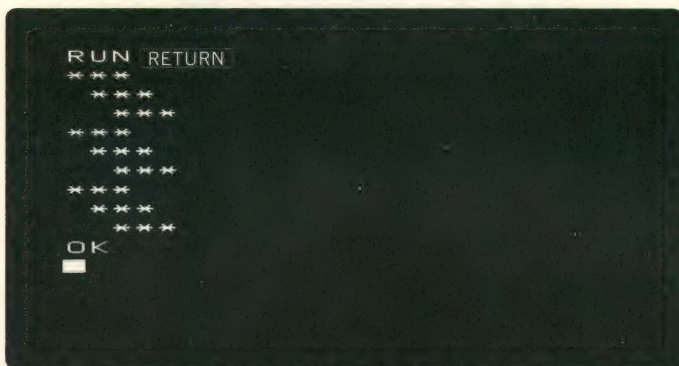
次のプログラムの中に，線で囲んだ空白がありますよネ。そこに，15文字以内の命令を入れて，プログラムをRUNしたときの画面が写真みたくなるように，してください。

FOR－NEXT命令の仕組みを，よーく考えてみると分ると思います。でも，コレは難しいですよー。

```
10 FOR I=0 TO 10 STEP 2
20 PRINT I;
30 IF I=
40 NEXT
50 END
```

第2問

今度のプログラムを，また空白の部分に命令を入れてください。このプログラムでは，IF文の代わりをどーやってすればいいのか，考えてくださいネ。13字以内です。

```
10  X=0
20  FOR Y=1 TO 9
30  LOCATE X,Y:PRINT "***";
40  X=[          ]
50  NEXT
60  END
```

答えは，残念ながら，みなさん自身で考えてくださいネ。

どーゆー答えでも，画面と同じ結果になれば，正しい答えなんです。自分でプログラムを作れば，思ったよりもカンタンにできます。

スタート
あなたはパソコンを持っていますか？
はい
いいえ
パソコンは，どんな機種を持ってますか？
①PC8801
②その他
①
②
PC8801の他にX1やFMなどのマイコンを持っていますか？
C言語というのを知ってますか？
ゲームで遊ぶ方よりも，外で遊ぶ方が好き？
ゲームで遊ぶのと，プログラムを作るのでは，プログラムを作る方が好き？
アニメをよく見ますか？
パラダイムとかダダとかを知ってますか？
月にいっぺんはパソコン・ショップへ行きますか？
オーディオとかビデオが好き？
どちらかといえば，ふとっている方ですか？
外へ着てゆく服は，気になりますか？
リン・ミンメイを知ってますか？
本は，よく読む方ですか？
BASICでプログラムを作るより，マシン語の方がラク？
インターフェースをよく読みますか？
プログラムを集めたディスケットをたくさん持ってますか？
MSXを持ってますか？
ASCIIを読んでますか？
A
B
C
D

A ふつうの少年	あなたは，ごくごく普通の少年です。あまりパソコンには向いてないかも。深入りするのは，危険です。
B パソコン少年	あなたは，とてもパソコンに興味を持ってる少年ですね。でも，技術力はまだまだ…。そのままシュミとして読けましょう。
C パソコン小僧	あなたは，なかなかの技術と自信を持った少年ですね。パソコン界でも十分に通用しますが，パソコン馬鹿には全然かないません…。
D パソコン馬鹿	あなたは，パソコン界でも1000人に1人の技術力を持っていますね。1人でパソコン界を変えることもできます。

おしらせ

みなさんの周りには，パソコンに興味を持ってる人って，たくさんいますー？　うーん，たぶん。あんまりいない，と思うんです。

おなじものに興味をもってるお友だちが少ないのって，なんだか，とってもさみしいですよネ。それに，なかなか上手になれなくって…。

そこで！　わたしたち2人は，パソコンクラブを作ろうかな？，なんて考えてるんです。**ファミコン**と**MSX**と**PC8801**を使ってる人のためのクラブですヨ。

会報を毎月おくる『会報中心』のクラブ。なまえは，えーと『**パソコン・倶楽部**』にしよーかな，なんて…。

わたしたちは受験なので，来年の春頃（86年の春）にはじめたいナ。で，もしよかったら，みなさんも入りませんか？　くわしいことは86年3月ごろのパソコン雑誌の通信欄に，出すつもりです。もし，良い考えがあったら『ナツメ社編集部』にハガキで出してくださいネ。ちゃんと，わたしたちに届きます。

このクラブは，会報中心にしようと思ってます。
でも，会報中心ってわけじゃないんです。3ヶ月に1度くらいは，みんなで集まってミーティングしたり…。
たまには，みんなでスケートに行ったりできたらいいなー。
内容は，オリジナルなゲームとか，入門とか。会員全員で，作ったり，考えたりするクラブにしたいですネ。

第3章 サンプルゲーム集です

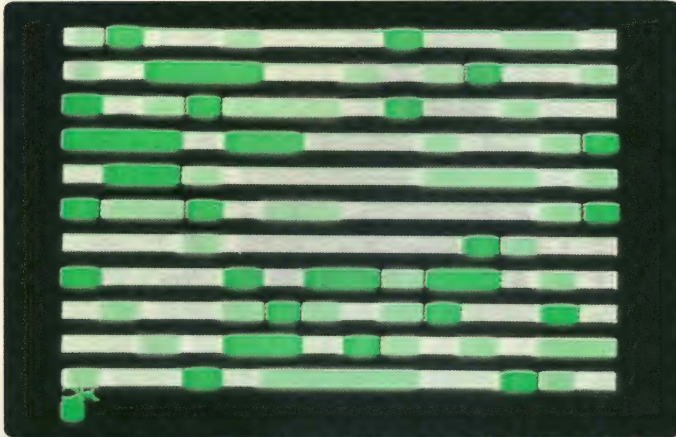
COLOR文と乱数を使ったサンプル

1 カラー・テスト

●REM

プログラム文にコメントを入れます。

説明

プログラムにコメントを挿入するためのステートメントで，実行にはかかわりありません。プログラム中どこでも書けます。REMは〈 ' 〉でも代用できます。REM 文で書かれた行番号は，何も実行されません。ただし，プログラム用メモリは使用されます。

```
100  '■SAMPLE-01■
150  PALETB 0,13,  1,13,13    ← 青
160  PALETB 1,13,  5,13,13    ← 赤
170  PALETB 2,13,  9,13,13    ← 緑
180  PALETB 3,13,17,13,13     ← 黄
200  FOR Y=0 TO 20 STEP 2
210  FOR X=0 TO 26 STEP 2
220  COLOR X,Y,RND(4):LOCATE
X,Y:PRINT CHR$(253);CHR$(253
);
230  NEXT
240  NEXT
300  END
```

このサンプルプログラムは，テレビ画面に**青色，赤色，緑色，黄色**の4色で〈■〉を表示します。**PALET文**を中心に，**LOCATE文，COLOR文**を使ってみました。

PALET文を使って，配色番号が⓪のときは青(色コード→1)，配色番号が1のときは赤(色コード→5)，配色番号が2のときは緑(色コード→9)，配色番号が3のときは黄(色コード→17)になるようにしてます。

あとは，FOR～NEXT文を使って，画面の座標の位置を決めて。で，COLOR文で色指定，LOCATE文で座標指定をしています。

それほど難しいサンプルプログラムじゃない，です。ただ，COLOR文とLOCATE文は，かならずCOLOR文が先になるようにしてくださいね。でないと，うまく指定できないみたい…。

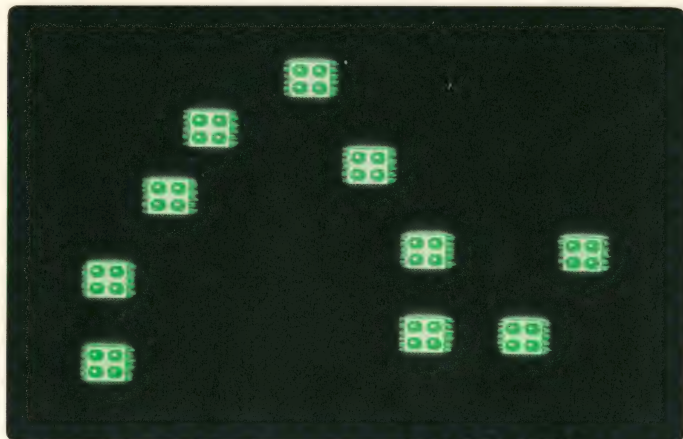
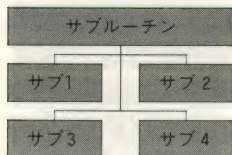
●PALET

配色番号内の色コードを任意の色コードに設定します。

背景の色やアニメキャラクタの表示色を，52色の色コードの中から選んで着色します。バックドロップの色は，バックグラウンドで使用しても，スプライトで使用しても画面に表示着色されます。プログラム上では実行行番号の大きなプログラムで指定した色が最終的に表示されます。

PALET文でサイコロがころころ転がります

2 ころころサイコロ

●**サブルーチン**
(subroutine)
プログラム全体の機能のうち論理的にまとまった一部を同行し，それを構成する特定の命令群が必要なときには，いつでも利用できるようになっているプログラムの一部。

```
100  '■SAMPLE-02■
150  VIEW
200  FOR I=1 TO 100
210   GOSUB 500
220   PALETB I MOD 4,13,48,48
, 5
230   FOR J=1 TO 200:NEXT ← 単なる時間待ちです
240  NEXT
250  END
260  '
500  '■SUB■
510   PALETB 0,13,13,13,13
520   PALETB 1,13,13,13,13 ← 全部を消します
530   PALETB 2,13,13,13,13
540   PALETB 3,13,13,13,13
550  RETURN
```

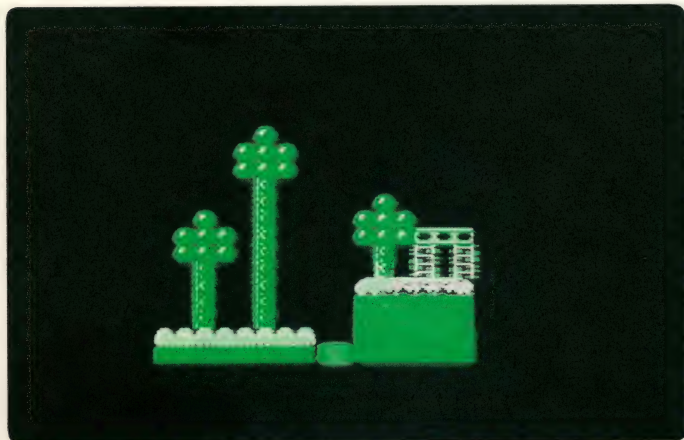
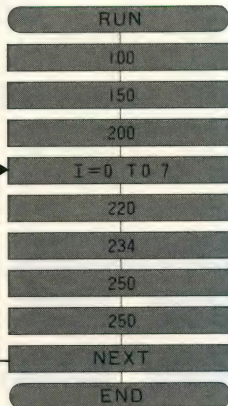
このサンプルは，サイコロがころころ転がるプログラムです。〈RUN〉でプログラムを動かす前に，**BG-GRAPHICで背景**を描いておいてくださいね。

このプログラムでは，**PALET文の変わった使い方**を，してます。

まず始めに，⓪〜3の配色番号でサイコロを描いておきます。そうしたら，配色番号=⓪のサイコロだけ，PALET文で白く設定します。次は，配色番号=1のサイコロを…。次は，配色番号=2のサイコロを…。すると，サイコロが動いてみえるんです。

ですから，動いてみえるサイコロは，実はぜんぜん動いてないわけです。ね，なかなか**不思議なサンプルプログラム**だ，と思いませんか？

マリオの散止の準備プログラムです

3 マリオの1・2・散歩(A)

●サンプル-02M

```
100  '■SAMPLE-02M■
150  SPRITE ON:VIEW
200  RESTORE
210  FOR I=0 TO 7
220    READ C,D,X,Y
230    IF D=0 THEN DEF SPRITEI
,(I MOD4,1,0,0,0)=CHR$(C)+CH
R$(C+1)+CHR$(C+2)+CHR$(C+3)
240    IF D=1 THEN DEF SPRITEI
,(I MOD4,1,0,1,0)=CHR$(C+1)+
CHR$(C)+CHR$(C+3)+CHR$(C+2)
250    SPRITE I,X*16+12,Y*16
260  NEXT
270  END
280  '
400  '■DATA■
410  DATA 0,1,2,10,4,1,3,10
430  DATA 8,1,4,10,12,1,5,9
460  DATA 4,1,6,9,0,1,7,9
480  DATA 20,1,8,9,20,0,8,8
```

●インタプリタ
(interpreter)
プログラムをそれが記述されている言語の構文上の基本単位(通常はステートメント)ごとに，解釈し，すぐにその場で実行するプログラム言語の一方式。シミュレーション言記やISPは，インタプリタであることが多い。

このプログラムは，『マリオの1・2・散歩』の準備のためのものです。プログラムを実行するまえに，BG-GRAPHICで背景を描いておきます。

『ころころサイコロ』が気に入ってました。なので，これをスプライトを使ったら面白そう！，なんて思って作ってみました。

プログラムをRUN命令で実行すると，**背景の森**と4色で色分けされた**マリオ**が，表示されます。これで実行終了です。ごくろうさま！

このあと，次のページの『マリオの1・2・散歩』を入力してくださいね。

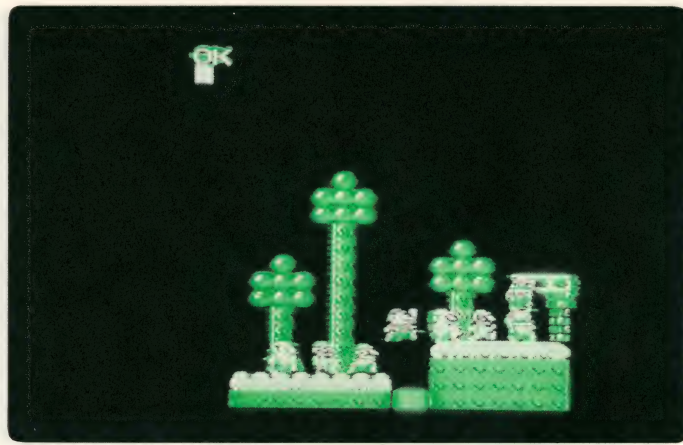
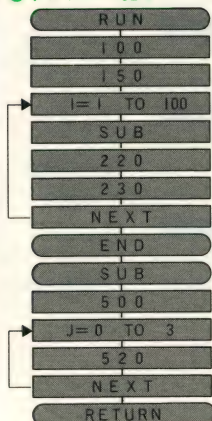
●READ

DATA で用意されたデータを READ 文の変数に入力します

READ 文は DATA 文に文して使うステートメントで，DATA のあとにはそれに対応するデータを書きます

マリオがトコトコ散歩をします

4 マリオの1・2・散歩(B)

●サンプル―02 S

```
100  '■SAMPLE-02S■
150  VIEW
200  FOR I=1 TO 100
210   GOSUB 500
220   PALETS I MOD4,13,54,22,
2
230   FOR J=1 TO 800:NEXT
240  NEXT
250  END
260  '
500  '■SUB■
510  FOR J=0 TO 3
520   PALETS J,13,13,13,13
530  NEXT
540  RETURN
550  '
```

(230行の NEXT ← 時間待ち)

●ステートメント
ステートメント(statement)は文とも呼ばれる．プログラムの言語の命令，または注釈ステートメント，翻訳操作を制御する指示文を含めて，コンパイラに入力できる任意の表現をさすのに使用することもある

マリオが，森の中を散歩するプログラムです。このプログラムを動かす時は，『マリオの1・2・散歩(A)』を **RUN命令で実行しておいてください。**

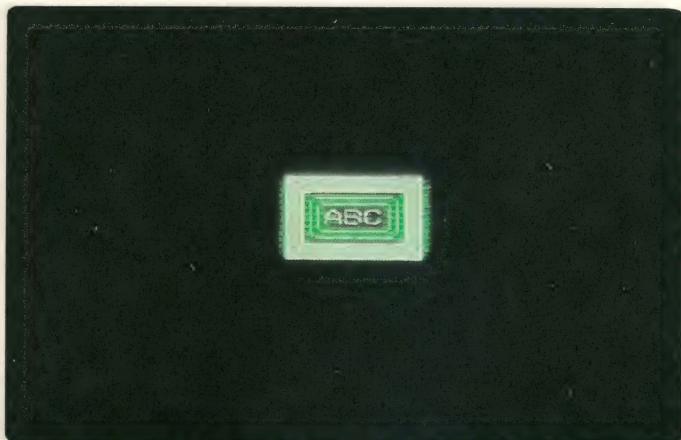
それから，このプログラムを実行するときは，

GOTO 100 (RETURN)

と入力します。RUN命令を使うと，『マリオ…A』で表示させておいた**スプライトが，消えちゃいます。**

500行から550行までのサブルーチンは，配色番号1～3の全部の色コード13（黒）にするためのもの，です。こうして全部のマリオを消しておいて，220行でひとつだけマリオを表示するようにしてます。

文字列を上手に使った簡単なスロットマシーン

5 スロットマシーン

●LOCATE

働き

テキスト画面のカーソルの位置を指定する

文法

LOCATE　X, Y

・X→水平方向の座標値
　Y→垂直方向の座標値

説明

LOCATEはバックグラウンド画面のカーソル位置を指定します。

```
100  '■SAMPLE-03■
150  VIEW
210  I$=CHR$(RND(26)+65)+CHR$
(RND(26)+65)+CHR$(RND(26)+65
)
220  IF  INKEY$="  "  THEN  GOSUB
 500
230  LOCATE  12,10:PRINT  I$
240  GOTO  210
250  '
500  '■SUB■
510  IF  INKEY$="  "  THEN  510
520  RETURN
```

(220行の500へ) スペース が押されていたら〈SUB〉へ

これは，スロットマシーンみたいなサンプルプログラム…です。このプログラムを実行（RUN命令を使って）する前に，BG-GRAPHICで背景を描いておいてくださいね。あッ，でも**背景が無くても大丈夫です。**

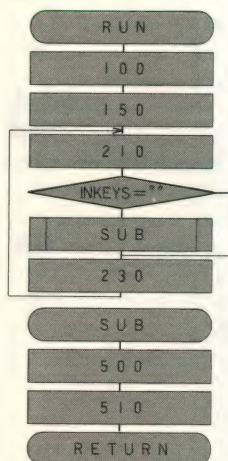
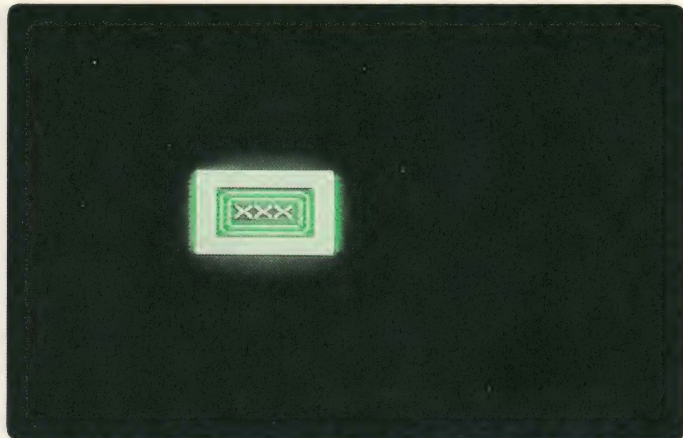
実行すると，3文字のアルファベットがパタパタ変わっていきます。何もしないと，このままズ〜ッと動きっぱなし…。

そこで スペース キーを押すと，パッと止まります。けれども，本物のスロットマシーンのように，文字が揃ってもコインが出てこないです。うーん，残念…です。

このプログラムでは，**RND関数の使い方**をみなさんに分かって欲しいな，と思って作ってみました。

220行と510行，**IF　INKEY$=" "　THEN**の使い方に注目してくださいね。

●サンプル－03

配列を上手に使ったスロットマシーン

6 ウルトラスロットマシーン

```
100  '■SAMPLE-04■
150  VIEW
160  DIM  A$(3)
200  FOR  I=1  TO  3
210   GOSUB  500
220  NEXT
250  IF  INKEY$=" "  THEN  X=X+1
:IF  X=5  THEN  X=1
260  LOCATE  12,10:PRINT  A$(1)
+A$(2)+A$(3);
270  GOTO  200
280  '
500  '■SUB■
510   IF  X>I  THEN  RETURN
520   A$(I)=CHR$(RND(26)+65)  ←文字を決めてます
530  RETURN
```

●配列
番号で区分けされた一画の記憶場所を配列(array)という。配列の中の個々の記憶場所は配列要素とか添字付変数と呼ばれ，添字値によってそれぞれが区別される。単純変数と比較して複雑な処理を行うことが可能となる。

前の『スロットマシーン』は，(SPACE)を押すと3文字が一緒にストップ，しました。これじゃあ，まだまだ本物らしくないな。

そんなふうに思ったので，**1文字ずつストップするように**してみました。

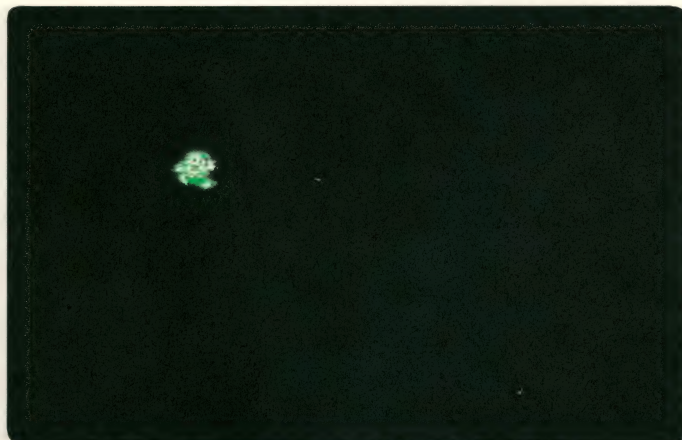
プログラムを動かす前には，やっぱりBG－GRAPHICで背景を描いた方が，楽しいッ，と思います。背景は『スロットマシン』と同じです。

このプログラムは，**配列に注目**して欲しい，です。

配列A＄（I）には，スロットマシーンの文字が入ってます。Aを（1）には左の，A＄（2）はまん中の，A＄（3）は右の文字が入っています。

この仕組みを作るのに，すごく考えました。

STICK文を使ってマリオを動かします

7 マリオをコントロール

●STICK
コントローラからの入力値を与えます
[illegible]
STICK (X)

```
100  '■SAMPLE-Ø5■
150  VIEW:SPRITE  ON
160  X=15Ø:Y=1ØØ
200  I=STICK(Ø)
210  IF  I=  Ø  THEN  M=Ø
220  IF  I=  8  THEN  M=1
230  IF  I=  9  THEN  M=2
240  IF  I=  1  THEN  M=3
250  IF  I=  5  THEN  M=4
260  IF  I=  4  THEN  M=5
270  IF  I=  6  THEN  M=6
280  IF  I=  2  THEN  M=7
290  IF  I=1Ø  THEN  M=8
300  DEF  MOVE(Ø)=SPRITE(Ø,M,1
,5,Ø,Ø)
310  POSITION  Ø,X,Y
320  MOVE  Ø←――――マリオを表示します
330  IF  MOVE(Ø)=-1  THEN  33Ø
340  X=XPOS(Ø):Y=YPOS(Ø)
350  GOTO  2ØØ
```

これは，マリオをみなさんが自由自在に動かせる，サンプルプログラムです。えと，まあ『**ゲームの基本**』というところ…。

このプログラムでは，**STICK関数**の使い方と，あとはXPOS関数，YPOS関数を使った**MOVE文**の利用の方法について，みなさんに分ってもらえたら…と思います。

MOVE文を使ったキャラクタの動し方を知りたい人は，300行のPOSITON文を消してみたり，330行のYPOS関数，YPOS関数を消してみるといい，と思います。こんなふうに，自分で変えてみることは，大切なことみたい…ですね。

・X→0，1
[illegible]
コントローラの値を与えます　対応する値は次の通りです　ただし何も押さない状態では0を与えます

関係演算子を使ったマリオコントローラ

8 マリオを簡単にコントロール

●関係演算子
関係演算子は2つの数値を比較するときに用いる。結果は真(－1)，偽(0)で得られ，条件判定文などのプログラムの流れを変えるのに利用される
「Hu-BASIC V2」では6種類の演算子が用意されている（ファミリーベーシック取扱説明書P51）

```
100  '■SAMPLE-06■
150  VIEW:SPRITE ON
160  X=150:Y=100
200  I=STICK(0)
210  M=-(I=8)*1-(I=9)*2-(I=1)
*3-(I=5)*4-(I=4)*5-(I=6)*6-(
I=2)*7-(I=10)*8
300  DEF MOVE(0)=SPRITE(0,M,1
,5,0,0)
310  POSITION 0,X,Y←――表示する位置を決めます
320  MOVE 0
330  IF MOVE(0)=-1 THEN 330
340  X=XPOS(0):Y=YPOS(0)
350  GOTO 200
```

これらの関係演算子は普通IF文の条件に使われる
たとえば次のように，である
・IF X＝0 THEN
・IF A＞1 THEN
このような条件演算は，その条件の真(－1)偽(0)によってその値が決まる。これを利用し，条件演算子を組み込んだ算術演算が可能となる
A＝10：PRINT
A＝10 (RETURN)
－1
OK
■

これは，前の『マリオをコントロール』と同じ内容のサンプルプログラム…になってます。

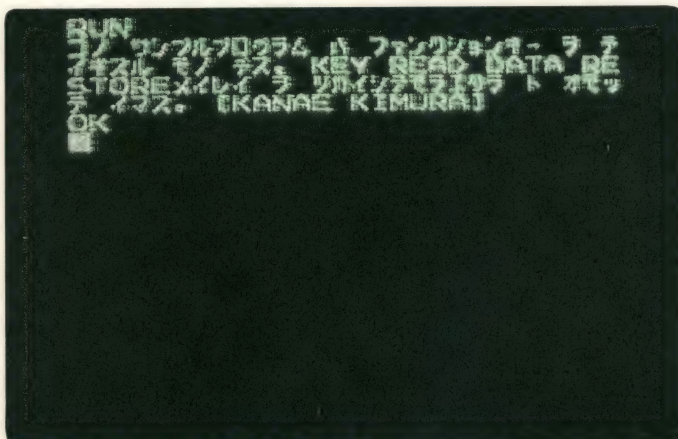
「同じ内容なら，また載せなくてもいいのに」。うーん，そんなふうに言われると，私困ってしまいます。でも，理由があるんです。

よく，2つのプログラムを比べてみてくださいね。そうすると，前のプログラムの210行から290行が，これでは210行ひとつだけになってます。これは，**関係演算子**にIF文のかわり，してもらったからです。

関係演算子は，**ちょっとしたパズル**です。

カッコの中の条件が正しいと，－1の値になるんだそうです。たとえば，変数1の値が4とします。すると（1＝4）の値が－1となり，X－5をした値が変数Mに入るんです。

KEY文を使ったサンプルプログラム

9 ファンクションめっせーじ

●KEY

ファンクションキーに対して文字列を定義します

KEY ファンクションキー番号, 文字列

```
100  '■SAMPLE-08■
150  RESTORE
160  FOR I=1 TO 8
170   READ I$:KEY I,I$
180  NEXT
190  '
200  '■DATA■
210  DATA " コノ サンプルプログラム ハ
220  DATA " ファンクションキー ヲ
230  DATA " テイキスル モノ デス。
240  DATA " KEY READ DATA
250  DATA " RESTOREメイレイ ヲ
260  DATA " リカイシテモラエタラ
270  DATA " ト オモッテ イマス。
280  DATA " [KANAE KIMURA]
```

FKEYに入れたいデータ

ファンクションキーに，メッセージを定義するサンプルプログラムです。このプログラムは，RUN命令で実行しただけだと，何の反応もないです。

そこで次のように入力してみてください。

F1	F2	F3	F4	F5	F6	F7	F8

するとメッセージが画面に表示されます。

DATA文の**データに文字列を使うとき**，気を付けた方がいいことが，あります。文字列をダブルクォーテーション〈"〉で囲むとき，その行の最後のダブルコーテーションは取った方がいいみたいです。行の最後にダブルコーテーションがあると，うまくRED文が動きません。(V2.0の場合)

●ファンクションキー番号
→1～8／文字列→最大15文字まで定義することができます

ファンクション番号で指定したキーに文字列を対応させます。このステートメントによって文字列を定義すると，それ以降そのキーを押すと定義した文字列をキーボードから打ち込むのと同じになります。

LOCATE文とPRINT文で積み木を動かす

10 積み木ワープ

●CHR$

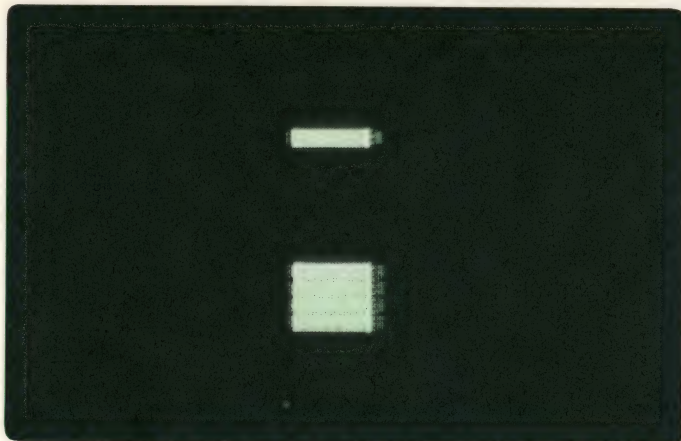
数値をキャラクタコードとみなして，対応する文字に変換します

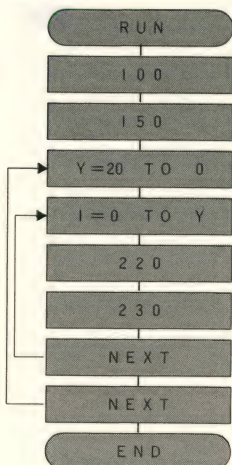
文法

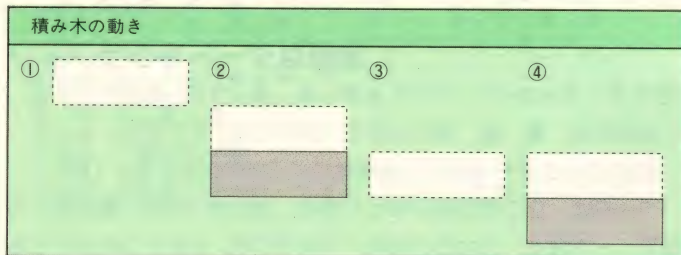
CHR$（X）

・X→文字に変換する数式

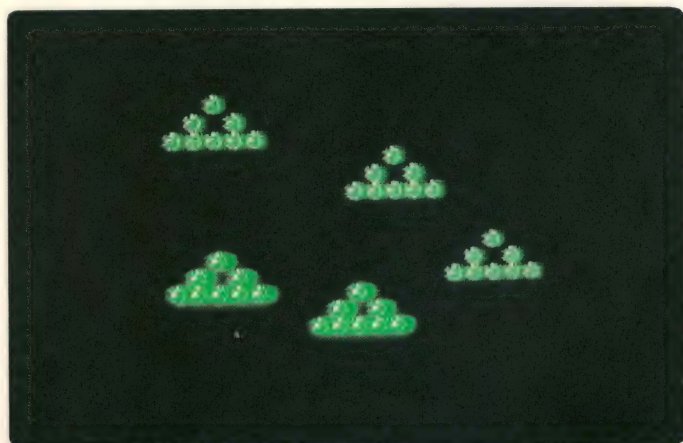
説明

数値をキャラクタコードとする文字が与えられます。数値1つにつき1文字が得られます。

```
100  '■SAMPLE-10■
150  VIEW
200  FOR Y=20 TO 0 STEP-1
210   FOR I=0 TO Y
220    LOCATE 10,I   :PRINT"
   "←積み木を消します
230    LOCATE 10,I+1:PRINT CH
R$(246)+CHR$(246)+CHR$(246)+
CHR$(246)←積み木を表示します
240   NEXT
250  NEXT
300  END
```

●サンプル①

これは，積み木がどんどん重なっていくサンプルプログラムです。

プログラムを実行すると，画面の上から「積み木」が落ちてきます。積み木が下まで落ちると，次のが落ちてきます。こうして，どんどん積み重なっていきます。

見てると本当に動いてるように見えます。でも，実は動いてるわけじゃないです。この仕組み，PRINT文でゲームを作るとき，とっても重要になります。

カメの子グラフィック

11 タートルグラフィック1

●RESTORE

READ文で読むDATA文を指定する

RESTORE〔行番号〕・行番号→読み始めるDATA文の行番号

行番号が省略されると次にくるREAD文はプログラム中の最初のDATA文から読み始めます。行番号を指定すると，指定された行のDATA文から読み始めます。

```
100  '■SAMPLE-11■
150  PALETB 1,13,48,39,10
160  PALETB 3,13,48,54,40
200  FOR I=1 TO 10
210   X=RND(24)+2  ┐
220   Y=RND(19)    ┘──表示する位置と色を決めてます
230   C=RND(4)
300   RESTORE
310   FOR J=1 TO 9
320    READ M,N
330    X=X+M:Y=Y+N
340    COLOR X,Y,C:LOCATE X,Y
:PRINT CHR$(215);
350   NEXT
360  NEXT
400  END
410  '
500  '■DATA■
510  DATA  0, 0, 1, 1, 1, 1
520  DATA -1, 0,-1, 0,-1, 0
530  DATA -1, 0, 1,-1, 1, 0
```

『ファミリーベーシック』で **簡単なタートルグラフィック**をやってみたくて，作ってみました。プログラムを実行すると，三角形が赤，青，緑，黄の4色で，次々と表示されます。

500行からのデータは，三角形をつくるためのデータ，です。

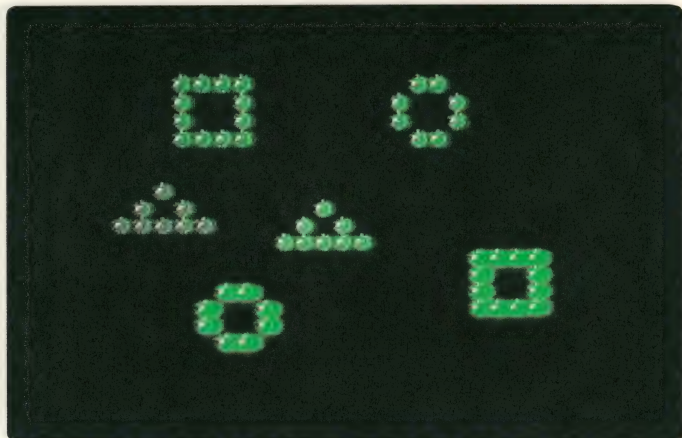
データは，2つずつ組になってます。前のデータは，横の動く方向で，－1は上，0はそのまま，1は下です。この2つのデータを組み合わせて，**三角形を描いてます。**

なかなかの出来だと，思っています…。

カメの子グラフィック その2

12 タートルグラフィック2

●ON GOSUB

働き

指定されたいくつかの行に分岐する

文法

ON 式 GOSUB 行番号［，行番号…］

・式→1から始まる数値変数 行番号→分岐する行の行番号

説明

式の値がプログラムのどの行番号の行に分岐するかを決定します。行番号の並びは1から始まる数に対応します。たとえば，式の値が3であったなら，行番号の並びの3番目の行へ分岐します。

```
100  '■SAMPLE-12■
150  PALETB 1,13,48,39,10
160  PALETB 3,13,48,54,40
200  FOR I=1 TO 10
210    X=RND(22)+2
220    Y=RND(18)
230    C=RND(4)
240    A=RND(3)+1
250    ON A GOSUB 710,720,730
310    FOR J=1 TO 13
320      READ M,N
330      X=X+M:Y=Y+N
340      COLOR X,Y,C:LOCATE X,Y
:PRINT CHR$(215);
350    NEXT
360  NEXT
400  END
410  '
500  REM*DATA*
510  DATA 0,0,1,0,1,1,0,1,-1,
1,-1,0,-1,-1,0,-1,1,-1,0,0.0
,0,0,0,0,0
520  DATA 0,0,1,0,1,0,1,0,0,1
,0,1,0,1,-1,0,-1,0,-1,0,0,-1
,0,-1,0,-1
530  DATA 0,0,1,1,1,1,1,1,-1,
0,-1,0,-1,0,0,-1,0,-1,0,0,0,
0,0,0,0,0
540  '
700  '■SUB■
710  RESTORE 510:RETURN
720  RESTORE 550:RETURN
730  RESTORE 600:RETURN
```

このプログラムは，三角形と四角形。それから，まる，をタートルグラフィックで描くものなんです。見た目がキレイなサンプルができました…。

SWAP文を使ったソート

13 大きい順に並べ！

●ソーティング
（sarting）
データの並びがあるとき，キーとして指定された属性の値がたとえばアルファベット順に並ぶように，各データを並べかえること。基本的なデータ処理のひとつであり，主記憶内のみで可能か否かなどの場合に応じて，種々の効率的なアルゴリズムが考えられる。

```
100  '■SAMPLE-13■
150  DIM D(9)
200  FOR I=0 TO 9
210    PRINT "データー";I;
220    INPUT D(I)
230  NEXT
300  FOR I=0 TO 8
310    FOR J=0 TO 8-I
320      IF D (J)<D(J+1) THEN SW
AP D(J),D(J+1)
330    NEXT
340  NEXT
350  FOR I=0 TO 9
360    PRINT D(I)
370  NEXT
400  END
```

（350〜360行）──並び替えたデータを表示します

このプログラムは，**数値の並べ変えをする**サンプルです。大きい順に並び替えるように，作りました。こんなふうに，数値や文字列の並び替えをすることを，**ソート**とか**ソーティング**と言うそうです。

プログラムを実行すると，「データー0？」と聞いてきます。そこで，なんでも好きな数値を入力してくださいね。

好きな数値（−32768〜＋32767）を10コ，入力します。

そうすると，入力した10コの数値が**大きい順に並べかえて**，表示されます。ソートは，**ゲームの得点の順位**とか**成績順に並べる**ときなんかに，使います。

300行から340行でソートをしています。なぜ，こんなプログラムで大きい順番に並ぶんでしょうねエ？　ホント，不思議です。

●DIM
働き
配列変数の要素の大きさを指定し，メモリ領域に割り当てる
説明
配列の名前と次元数，および添字のサイズを設定します。１つのDIM文で複数の配列を宣言できて，配列ごとにメモリの範囲内で２次元までの添字の指定ができます。

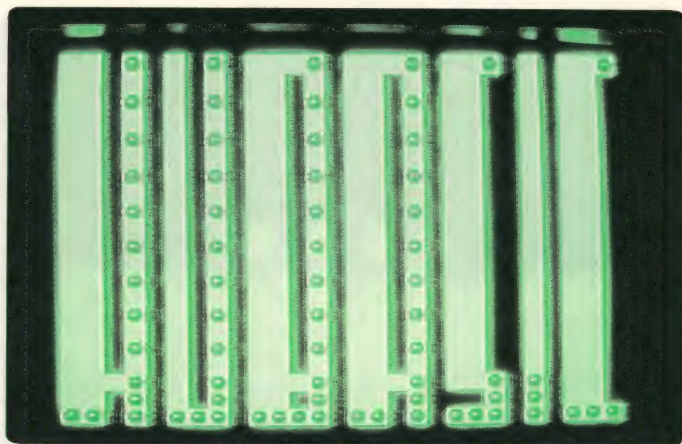
MOVE文でUFOが飛びまわります

14 ムズムズMOVE

●SPRITE ON

スプライト面の表示を可能にします，スプライト表示モード

SPRITE ON

スプライト面をバックグラウンド面に重ねて表示できる状態にします
スプライト面上にあるスプライトが，画面上に表示されます

```
100  '■SAMPLE-15■
150  VIEW:SPRITE ON
160  X=18*8:Y=10*8
200  DEF MOVE(0)=SPRITE(7,RND
(4)*2+2,1,RND(30)+1,1,0)
210  POSITION 0,X,Y  ↑動く距離  ↑動く方向
220  MOVE 0
230  IF MOVE(0) THEN 230
240  X=XPOS(0):Y=YPOS(0)
250  GOTO 200
```

●MOVE

アニメキャラクタの動きを開始させます。

MOVE n_0 [, n_1, n_2, n_3, n_4, n_5, n_6, n_7]

あらかじめDEF MOVE命令で定義されたアニメキャラクタの動きを開始させます。アニメキャラクタの動きを同時に8個まで開始させることができます

ちょっぴりゲームらしい サンプルプログラムを…，と思って作ってみました。これは，**背景** を作るのにとっても時間が掛ったんです。

このプログラムを実行する前に，BG-GRAPHICを使って背景を描いてください…ね。実行すると，背景のバックに，スピナーがクルクル回転しながら飛んできます。

200行の**DEF MOVE文** は，表示するスプライト（アニメキャラクタ）の動きを，種類，色なんかを決めてます。

214行の**POSITION文**は，スプライトの表示する場所を決めてます。

220行の**MOVE文**は，スプライトを動かす命令です。

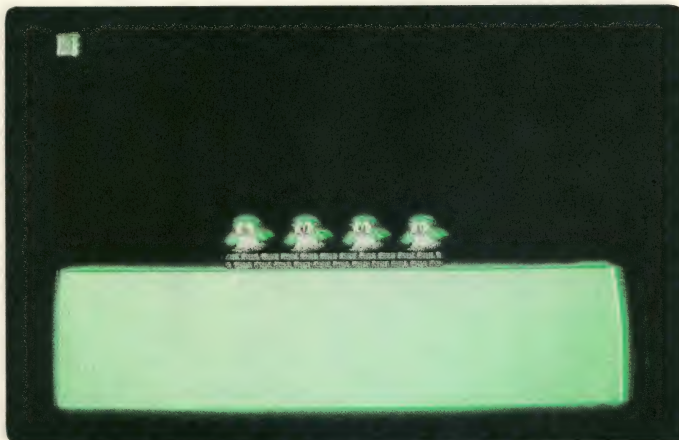
230行では，**MOVE関数**を使ってスプライトが動き終わったのかを，調べてます。

240行は，動き終わったスプライトの場所を，変数Xと変数Yに入れてます。

DATA文でペンギンが上手に踊ります

15 ペンギン・ダンス物語

●**プログラムについて**

この香奈枝嬢により作られたサンプルプログラムには，何の問題点もないように思われる。極めて優れた構造である。例えば220行，230行のサブルーチン処理である。このようにＩＦ文での処理が複雑になる場合，サブルーチン構造にするのが良い。またデータの扱い方にもなかなか優れた点がある。

```
100  '■SAMPLE-16■
150  CGSET1,2:VIEW:SPRITE ON
160  RESTORE
200  FOR I=0 TO 28
210   READ D
220   IF D MOD2=0 GOSUB 700
230   IF D MOD2=1 GOSUB 800
250   FOR J=0 TO 3
260    SPRITE J,J*24+80,102
270   NEXT
280   PAUSE 50
290  NEXT
300  GOTO 160
310  '
400  '■DATA■
410  DATA 104,105,104,105,104
,105,104,105,104,105,100
420  DATA  96,100, 96,108,109
,108,109, 97,101, 97,101
430  DATA 104,105,104,105,104
,105,104,105
440  '
700  '■SUB1■
710  FOR J=0 TO 3
720   DEF SPRITE J,(0,1,0,0,0
)=CHR$(D)+CHR$(D+1)+CHR$(D+2
)+CHR$(D+3)
730  NEXT
740  RETURN
750  '
800  '■SUB2■
810  FOR J=0 TO 3
820   DEF SPRITE J,(0,1,0,1,0
)=CHR$(D)+CHR$(D-1)+CHR$(D+2
)+CHR$(D+1)
830  NEXT
840  RETURN
850  '
```

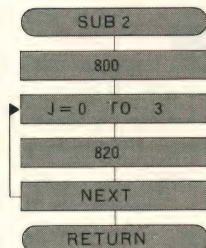
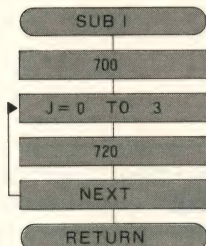
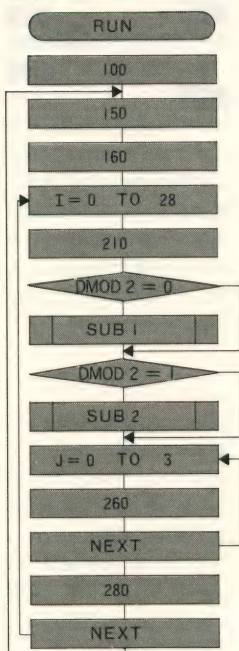
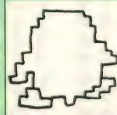
ペンギンのダンスのデータです

●CGSET

ＢＧやスプライトで使用するパレットの割り当てを決めます。

説明

あらかじめ用意されている色の組み合わせのグループの中から選んで，背景の色やアニメキャラクタの表示色を決めます。表示可能な全52色の中からバックグラウンド用としてスプライト用にカラーチャートで示されているように，それぞれのカラーパレット上へ色が設定されます。バックグラウンド用で２種類スプライト用で３種類のカラーパレットが準備されています。それぞれのカラーパレットには配色番号の各グループに各々３種類ずつ，計12色の色コードが保持できます

●サンプル-16

このプログラムは，**ペンギン君が4ヒキ揃ってダンスをする**サンプルです。本当は，このサンプル集は1ページに1プログラムのつもりでした…。なるべく面白いサンプルにしたいな，なんて思ったら1ページに入りきらなくなっちゃった。

プログラムを実行する前に，BG-GRAPHICを使って背景を入力しておいてくださいね。やっぱり，ペンギン君はちゃんと台の上で踊りたいもん。

RUN命令を実行すると，ペンギン君が4ヒキでダンスします。いつまでも，ずーっとダンスしてます。そこで，やめたい時には（STOP）キーを押します。

まだ，かなりスペースがあるので，400行からのDATA文のデータについて，説明したいと思います。

ペンギン君のキャラクタは，次のようになってます。

それで，DATA文の中の**データが100**だと，「100を2で割った余り＝0」なのでサブルーチン〈SUB1〉を呼びます。もしも，**データが101**だと，「100を2で割った余り＝1」だからサブルーチン〈SUB2〉を呼びます。

サブルーチン〈SUB1〉では，4ヒキのペンギン君をキャラクタテーブルと同じ形に表示するように，決めてます。でも，**サブルーチン〈SUB2〉**では，キャラクタテーブルと左右さかさまに表示するように，決めてます。

```
DEF SPRITE J,(0, 1, 0, 0, 0)=
  ・キャラクタテーブルと同じ形 ─┘  │
  ・キャラクタテーブルと同じ形 ────┘
DEF SPRITE J,(0, 1, 0, 1, 0)=
  ・キャラクターテーブルと同じ形 ─┘  │
  ・キャラクタテーブルと同じ形 ─────┘
```

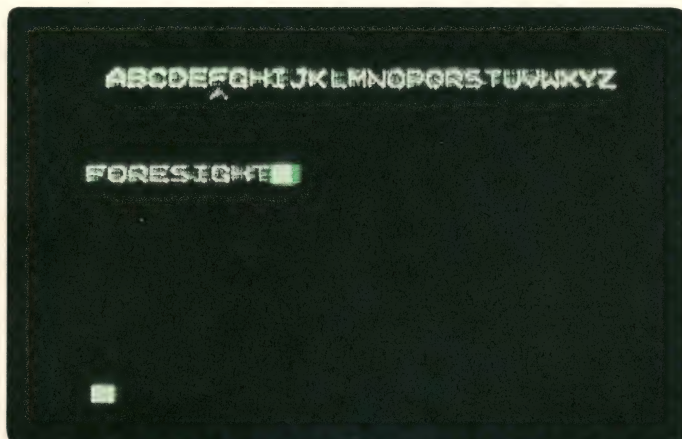
MID＄を利用した入力プログラム

16 手間なしキーボード入力

●MID＄

文字列の中から任意の長さの文字列を与える

説明

文字列中の開始位置から取り出された n 個の文字列がこの関数の値になります。開始位置が文字列の数より大きいときには，ヌルストリングが関数の値になります。

```
100  '■SAMPLE-17■
150  SPRITE OFF:CLS
160  D$="ABCDEFGHIJKLMNOPQRST
UVWXYZ":X=13:M=0:N=5
170  LOCATE 1,0:PRINT D$;
200  I=STICK(0)
210  IF I=2ANDX>0 THEN X=X-1
220  IF I=1ANDX<25 THEN X=X+1
240  LOCATE X,1:PRINT" ^ ";
250  IF STRIG(0)=8 GOSUB 500
260  LOCATE M,N:PRINT"■";
270  GOTO 200
280  '
500  '■SUB■
510  I$=MID$(D$,X+1,1)
520  LOCATE M,N:PRINT I$;
530  M=M+1:IF M=28 THEN M=0:N
=N+1
540  RETURN
```

入力する文字を選びます（510行）

●ヌルストリング

ヌルストリング（null-string）は何もない文字列のこと。文字式で表すと“ ”となる。BASIC起動直後，CLEAR命令，NEW 命令実行後がでて，文字変数はこの状態になる。

コントローラーを使ってアルファベットを入力するサンプルプログラムを，作ってみました。キーボードから入力するのが面倒だ，と思ってる人向き，ですね。

プログラムを実行すると，テレビ画面の一番上にA～Zのアルファベットが表示されます。その下に**アップアロー**〈^〉が表示されてます。

このアップアロー〈^〉を**コントローラで左右**に動かして，好きな文字の下でストップ。それで，**コントローラAボタン**を押します。すると，画面に選んだ文字が表示，されます。間違えちゃうと，もう消せません。注意，します…。

●STRIG

コントローラのトリガーボタンからの入力状態を与えます。

コントローラのトリガーボタンを押したときの値を与えます。コントローラ Ⅰ の場合，SELECT＝2，START＝1，A＝8，B＝4。コントローラ Ⅱ の場合，A＝8，B＝4です。

SCR$で画面の文字を調べます

17 CRTスキャナー

●CLS

働き

画面をクリアします。

説明

バックグラウンド面をクリアします。ただし，バックグラウンド面にコピーされたBG-GRAPHICも同時に消えます。プログラム上でBG-GRAPHICをバックグラウンド面にコピーする場合は，CLS命令を入れ替えてVIEW命令を使用してください。

```
100  '■SAMPLE-19■
120  CLS
130  FOR I=1 TO 50:LOCATE RND
(28),RND(23):PRINT CHR$(RND(
255));:NEXT
140  X=13:Y=12
200  LOCATE X,Y:PRINT" ";      ←カーソルを消します
210  I=STICK(0)
220  IF I=2 THEN X=X-1
230  IF I=1 THEN X=X+1
240  IF I=8 THEN Y=Y-1
250  IF I=4 THEN Y=Y+1
260  LOCATE 5,23:PRINT"キャラクタコ
ード";ASC(SCR$(X,Y));"  ";
270  LOCATE X,Y:PRINT"■";      ←カーソルを表示します
280  PAUSE 5
290  GOTO 200
300  '
```

●カラム(Column)

名 1▶円柱／2▶円柱のようなもの／3▶欄／4▶船の列，(数字の名前などの)縦の行

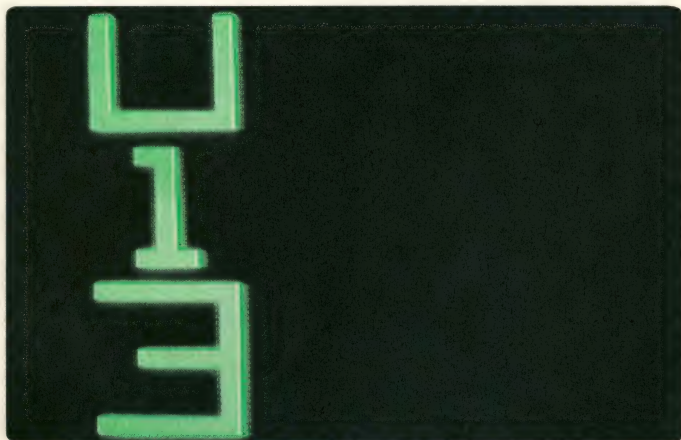
テレビ画面に表示されてる**文字のキャラクタコードを調べる**サンプルプログラムです。

プログラムを実行すると，画面の真ん中にカーソル〈■〉が表示されます。このカーソルは，コントローラ Ⅰ で動かせます。

こうやって，カーソル位置にある文字（キャラクタ）のキャラクタコードを調べることが，できます。

うーん，なんか物足りない感んじ…，なんて思ったり。もしも，そう思ったら，改良してみてください…ね。

このプログラムは，**SCR$関数**の使い方を分かってくれたらいいな，と思って作ってみました。

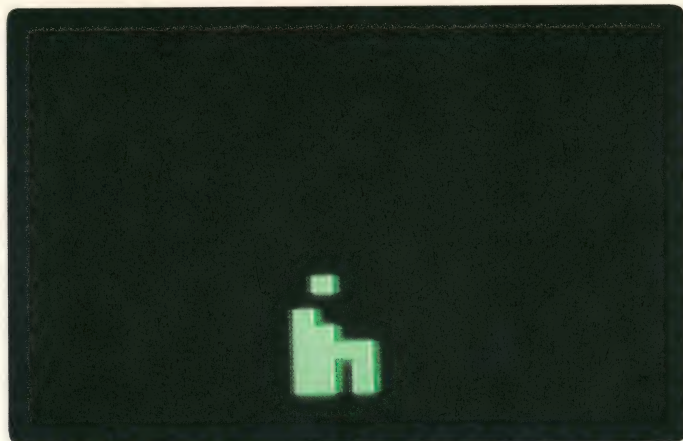
18 拡大数字データ入力

```
100  '■SAMPLE-21■
110  DIM D(8),D$(8),A(8)
120  D(1)=1:D(2)=2:D(3)=4:D(4
)=8:D(5)=16:D(6)=32:D(7)=64:
D(8)=128
200  D$(1)="          "
210  D$(2)="          "
220  D$(3)="          "
230  D$(4)="          "
240  D$(5)="          "
250  D$(6)="          "
260  D$(7)="          "
270  D$(8)="          "
300  FOR I=1 TO 8
310   FOR J=1 TO 8
320     IF MID$(D$(K),J,1)="O"
 THEN A(I)=A(I)+D(J)
330   NEXT
340  NEXT
400  PRINT"*16*    *10*"
410  FOR I=1 TO 8
420   PRINT HEX$(A(I)),A(I)
430  NEXT
500  END
```

『拡大数字のプログラム』のための，サンプルです。

この前のプログラムの710行〜750行のデータ。これは，数字の形のデータです。この数字データを作るためのプログラム，なんです。

プログラムを実行する前に，200行〜270行に表示したい文字の形を〈O〉で描いておいてくださいね。えと，文字の大きさは8×8の大きさになってます。このプログラムと前のプログラムは，ほとんどスタッフの方に聞いて作りました。

DATA文とREAD文でブロックが行動します

19 四神合体ブロック

```
100 '■SAMPLE-23■
110 RESTORE
120 FOR I=1 TO 17
130  READ X,Y
140  FOR J=0 TO X-1
150   LOCATE J,0:PRINT" ■"
160  NEXT
170  FOR J=1 TO Y
180   LOCATE X,J-1:PRINT" "
190   LOCATE X,J  :PRINT"■"
200   NEXT
210 NEXT
220 END
230 '
300 '■DATA■
310 DATA 11,20,12,20,11,19
320 DATA 11,18,11,17,12,19
330 DATA 14,20,12,18,11,16
340 DATA 14,19,14,18,13,18
350 DATA 12,17,12,16,14,17
360 DATA 14,16,13,16
```

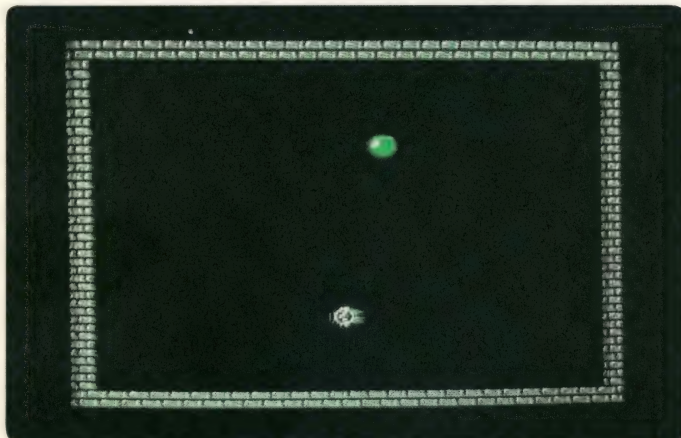
これは，**ブロックが飛んできて**〈A〉という字を作る，サンプルプログラムです。

プログラムを実行すると，ブロックが左上から飛んできます。そのブロックがだんだん積み重なって〈A〉という文字になります。

うーん，それほど難しいわけじゃないんですね。

データを変えて，Aの代わりにBを表示するようにしてみたり…。それから,〈■〉の代わりにレンガ〈□〉にしてみると，面白いと思います。でも，データを変えたら，120行の値も変えないと，うまく動かないです。

SGN(I)でエイリアンが追ってきます

20 追っかけエイリアン

●STICK

コントローラのボタンの入力値を与えます

STICK (X)

▶X…0, 1

コントローラのボタンの値を与えます。

```
100 '■SAMPLE-25■
110 VIEW
150 X=15:Y=5:M=13:N=15
200 FOR I=1 TO 1000
210  LOCATE X,Y:PRINT"
220  K=STICK(0)
230  IF K=2 AND X>1  X=X-1
240  IF K=1 AND X<26 X=X+1
250  IF K=8 AND Y>1  Y=Y-1
260  IF K=4 AND Y<19 Y=Y+1
270  LOCATE X,Y:PRINT CHR$(207)
300  IF I MOD 3=0 GOSUB 500
310 NEXT
320 END
330 '
500 '■SUB■
510 LOCATE M,N:PRINT" "
520  M=M+SGN(X-M)
530  N=N+SGN(Y-N)
540 LOCATE M,N:PRINT CHR$(180)
550 RETURN
```

みなさんの動かす〈●〉を，**エイリアンが追っかけてくる**サンプル・プログラムです。

プログラムを動かす前に，BG-GRAPHICを使って背景を描いておいてくださいね。動かすと，テレビ画面に〈●〉と〈©〉が出てきます。

それで，この〈●〉の方をみなさんが動かします。**コントローラを，使います。**

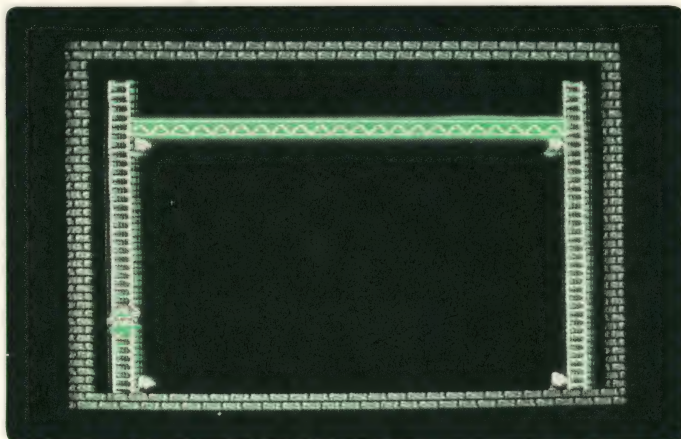
〈©〉はエイリアンのつもりです。エイリアンは，動く〈●〉を追いかけてくるんです…。

MOVE文でいろんな動きをします

21 うごめくマリオブラザーズ

●PAUSE

プログラムの実行を一時休止します

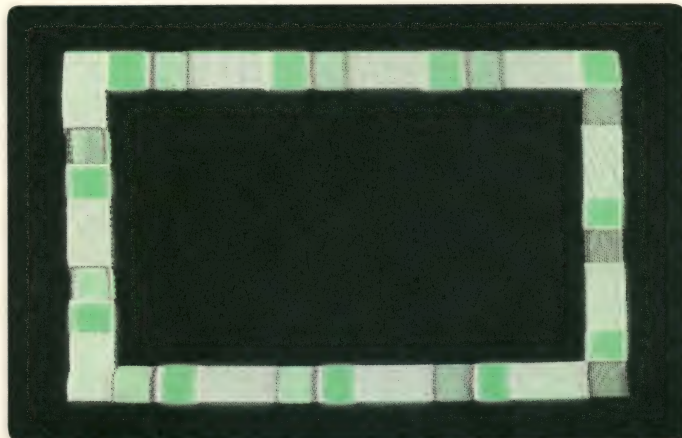
このステートメントを実行すると，設定単位時間分プログラムの実行を休止したのち次に進みます

```
100 '■SAMPLE-26■
110 CGSET ,2:VIEW:SPRITE ON
200 DEF MOVE(0)=SPRITE(0,7,1
,11*4)
210 POSITION 0,12*8+20,17*8+
24
220 MOVE 0:PAUSE 150
230 DEF MOVE(0)=SPRITE(0,1,2
,15*4)
240 POSITION 0,XPOS(0),YPOS(
0)
250 MOVE 0:PAUSE 400
260 DEF MOVE(0)=SPRITE(0,3,1
,23*4)
270 POSITION 0,XPOS(0),YPOS(
0)
280 MOVE 0:PAUSE 300
300 DEF MOVE(0)=SPRITE(0,5,2
,15*4)
310 POSITION 0,XPOS(0),YPOS(
0)
320 MOVE 0:PAUSE 400
330 DEF MOVE(0)=SPRITE(0,7,1
,14*4)
340 POSITION 0,XPOS(0),YPOS(
0)
350 MOVE 0:PAUSE 140
400 GOTO 200
```

マリオが地面を歩いたり，ハシゴを登ったりするサンプル・プログラムです。

プログラムを動かす前に，BG-GRAPHICで背景を描いておいてくださいね。これは，背景が描いてないと，ぜんぜん面白くないです。でも，一応，動きます…。

MOVE文の使い方を説明したくて，作りました。

PALET文を上手に使ったネオンサイン

22 ぐるぐる・ネオンサイン

●配列
配列とは変数の一種のことす。通常の変数は，１つの記憶場所に１つの役割が与えられています。しかし配列の場合には，１つの役割りに複数の記憶場所が必要な時に使用されます。例えばクラスの生徒の成績を記憶するなど。

```
100  '■SAMPLE-27■
110  DIM C(3)
120  VIEW:C=0
200  FOR I=0 TO 3
210   C(I)=(C+I)MOD 4
220  NEXT
250  PALETB C(0),13,0,0,48
260  PALETB C(1),13,0,0,42
270  PALETB C(2),13,0,0,38
280  PALETB C(3),13,0,0,32
290  C=C+1:IF C=4 THEN C=0
300  GOTO 200
```

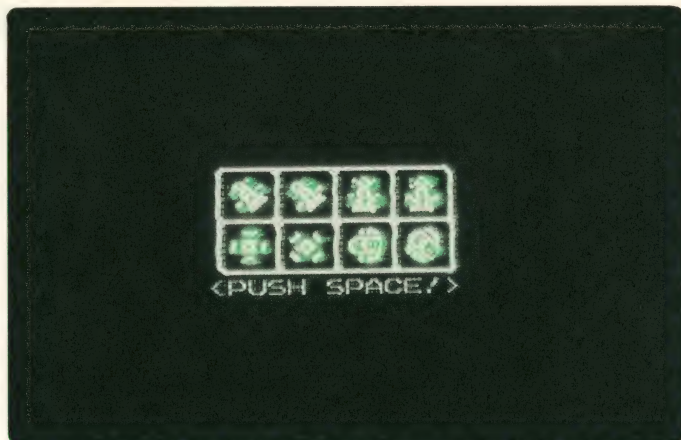
ネオンサインみたいなサンプル・プログラム…です。
プログラムを動かす前に，BG-GRAPHICを使って背景を描いておいてくださいね。それで，プログラムをRUN命令で動かすと，テレビ画面の周りがぐるぐるネオンサインみたいに回ります。なかなかキレイ，なんです。
こういうのって，よくテレビなんかで使われてるでしょ？　わァ，わたしもやってみたいナ，と思って作りました。
PALET文とMODを上手に使って，作ってあります。そうそう，**配列**も上手に使ってみました。
うーん，やっぱり200行から220行のとこが，とっても考えさせられちゃいました。本当は，もっとゴチャゴチャ複雑だったんです。
みなさんも，**どうしてグルグル回るのか**，考えてみてくださいね。

SPRITE文で，全スプライトを表示

23 スプライト表

●DEF SPRITE

スプライト面上に表示したいスプライト（アニメキャラクタ）を定義します。

スプライト面に表示したいアニメキャラクタを定義します。スプライト番号は，0～7の8コまで指定し，定義できます。キャラクタ合成型に0を指定すると，1キャラクタでアニメキャラクタを構成します。1を指定すると，4キャラクタで構成します。

```
100  '■SAMPLE-14■
150  VIEW:SPRITE ON
170  A=0
200  FOR I=0 TO 7
210    J=A+I*4
220    DEF SPRITE I,(0,1,0,0,0
)=CHR$(J)+CHR$(J+1)+CHR$(J+2
)+CHR$(J+3)
230    SPRITE I,80+(I MOD4)*24
,79+I/4*24
240  NEXT
300  IF INKEY$<>" " THEN 300
310    A=A+32
320    IF A>255 THEN A=0
330  GOTO 200
```

（170行 A=0：表示したいキャラクタの番号）

●SPRITE

働き

定義されているスプライトを任意の位置に表示または消去します。

説明

すでにDEF SPRITEで定義されているアニメキャラクタをスプライト面上の任意の位置に表示します。水平方向・垂直方向の座標を省略すると，指定のスプライト番号のアニメキャラクタを消去します。

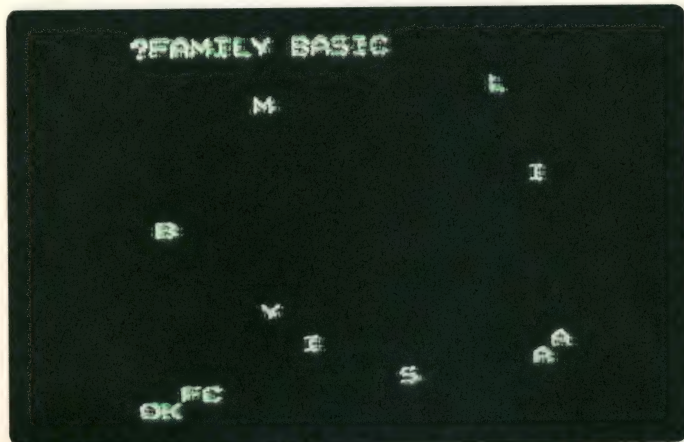
アニメキャラクタ（スプライト）の全部を，テレビ画面に表示するプログラムを，作ってみました。**SPRITE文**を，使っています…。

プログラムを実行する前に，BG-GRAPHICで背景を描いておいてください…ね。でも，面倒な人，背景を描かなくても動くから，そのまま実行してもいいです。なるべくなら，背景をキチンと描いてくれた方が，うれしい…。

実行すると，アニメキャラクタが8コ，テレビ画面に表示されます。なんだか，**スロットマシンみたい。**

このプログラム，本当は全部のアニメキャラクタをテレビ画面に表示してみたい，と思って作りました。けど，アニメキャラクタは同時に8コまで…，というのを忘れてました。

MID$とRUD(I)で単語をバラバラにします。

24 単語バラバラ殺人事件

●INPUT

キーボードから数値や文字を入力します。

キーボードから入力した数値を変数に入れます。あるデータを入力する場合，前もって入れる変数を用意しなければなりません。

変数は数値変数，文字変数のどちらでもかまいません。けれど，入力するデータと同じ型でなければいけません。

```
100  '■SAMPLE-28■
110  CLS
150  INPUT A$
200  FOR I=1 TO LEN(A$)
210   LOCATE RND(28),RND(21)
220   PRINT MID$(A$,I,1);
230  NEXT
240  END
```

みなさんが入力した単語を1つ1つバラバラにしてテレビ画面に表示するサンプルプログラムです。入力するのは単語じゃなくてもいいんです。文章でも，255文字より少なければ良いそうです。

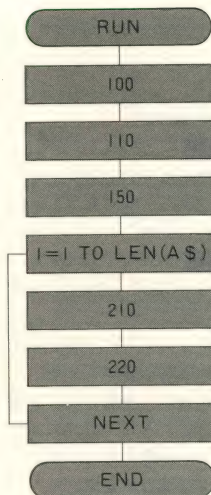
プログラムを動かすと，テレビ画面に〈?〉と出てきて表示する単語（文章でもいいんです）を聞いてきます。なので，キチンと答えてあげてください。

そうすると，入力した**単語がバラバラ**になって表示されます。

このプログラムは，LEN関数とMID$関数の使い方がみなさんに分ってもらえたら…と思って作りました。ぜんぜん難しくないので，よーく見てください。

LEN関数は，文字列の長さを調べるための関数です。200行では，文字変数のA$の長さ分だけ繰り返すようにしています。

MID$関数は，文字列からどこか一部分だけ取り出しちゃう関数なんです。220行では，みなさんが入力した単語から一文字だけ取り出してます。

●サンプル28

サンプル用BGグラフィック画面

●サンプル02

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
0																												
1							M11	M11	M12	M12																		
2							M11	D41	D42	M12																		
3					M10	D40					D43	M13																
4					M10	M10					M13	M13																
5					D43	M13					M10	D40																
6					M13	M13					M10	M10																
7			M12	M12									M11	M11														
8			D42	M12									M11	D41														
9			M11	M11									M12	M12							M11	M11						
10			M11	D41									D42	M12							M11	D41						
11			M10	D40									D43	M13					M10	D40			M12	M12				
12			M10	M10									M13	M13					M10	M10			D42	M12				
13	D43	M13													M10	D40			D43	M13			D43	M13				
14	M13	M13													M10	M10			M13	M13			M13	M13				
15	M12	M12													M11	M11	M12	M12							M10	D40		
16	D42	M12													M11	D41	D42	M12							M10	M10		
17	M11	M11													M12	M12	M11	M11							M11	M11		
18	M11	D41													D42	M12	M11	D41							M11	D41		
19	M10	D40													D43	M13	M10	D40							M12	M12	D43	M13
20	M10	M10													M13	M13	M10	M10							D42	M12	M13	M13

●サンプル02M

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
0																												
1																												
2																												
3																												
4																												
5																												
6																												
7																												
8									G71																			
9								G71	G71	G71																		
10								G71	G71	G71																		
11									H30																			
12						G71			H30					G71														
13					G71	G71	G71		H30				G71	G71	G71		H20	H10	H10									
14					G71	G71	G71		H30				G71	G71	G71		H20		F30									
15						H30			H30					H30			H20	H10	F30									
16						H30			H30					H30			H20		F30									
17						H30			H30				I03	I13	I13	I13	I13	I13	I13									
18						H30			H30				H53	I43	I43	I43	I43	I43	I43									
19			I13	I13	I13	I13	I13	I13	I13	I23			H53	I43	I43	I43	I43	I43	I43									
20			I43	I43	I43	I43	I43	I43	I43	I43	M70	M70	I43	I43	I43	I43	I43	I43	I43									

●サンプル3

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
0																												
1																												
2																												
3																												
4																												
5																												
6																												
7																												
8											L30	L60	L60	L60	L60	L60	L70											
9											L40	I60	J30	J30	J30	I70	L40											
10											L40	J20				J20	L40											
11											L40	J00	J30	J30	J30	J10	L40											
12											L50	L60	L60	L60	L60	L60	M00											
13																												
14																												
15																												
16																												
17																												
18																												
19																												
20																												

●サンプル14

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
0																												
1																												
2																												
3																												
4																												
5																												
6								K70	J30	J30	K20	J30	J30	K20	J30	J30	K20	J30	J30	L04								
7								J20			J20			J20			J20			J20								
8								J20			J20			J20			J20			J20								
9								K40	J30	J30	K00	J30	J30	K00	J30	J30	K00	J30	J30	K30								
10								J20			J20			J20			J20			J20								
11								J20			J20			J20			J20			J20								
12								L10	J30	J30	K10	J30	J30	K10	J30	J30	K10	J30	J30	L20								
13																												
14								〈	P	U	S	H		S	P	A	C	E	!	〉								
15																												
16																												
17																												
18																												
19																												
20																												

●サンプル15

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
0	M60	M60		D40		M60		D40		M60	M60	M60	D40		M60	M60	M60	D40		M60	M60	D40		M60		M60	M60	D40
1	M60	M60		M60		M60		M60		M60	M60		M60		M60	M60		M60		M60	M60			M60		M60	M60	
2	M60	M60		D40		M60		D40		M60	M60		D40		M60	M60		D40		M60	M60			M60		M60	M60	
3	M60	M60		M60		M60		M60		M60	M60		M60		M60	M60		M60		M60	M60			M60		M60	M60	
4	M60	M60		D40		M60		D40		M60	M60		D40		M60	M60		D40		M60	M60			M60		M60	M60	
5	M60	M60		M60		M60		M60		M60	M60		M60		M60	M60		M60		M60	M60			M60		M60	M60	
6	M60	M60		D40		M60		D40		M60	M60		D40		M60	M60		D40		M60	M60			M60		M60	M60	
7	M60	M60		M60		M60		M60		M60	M60		M60		M60	M60		M60		M60	M60			M60		M60	M60	
8	M60	M60		D40		M60		D40		M60	M60		D40		M60	M60		D40		M60	M60			M60		M60	M60	
9	M60	M60		M60		M60		M60		M60	M60		M60		M60	M60		M60		M60	M60			M60		M60	M60	
10	M60	M60		D40		M60		D40		M60	M60		D40		M60	M60		D40		M60	M60			M60		M60	M60	
11	M60	M60		M60		M60		M60		M60	M60		M60		M60	M60		M60		M60	M60			M60		M60	M60	
12	M60	M60		D40		M60		D40		M60	M60		D40		M60	M60		D40		M60	M60			M60		M60	M60	
13	M60	M60		M60		M60		M60		M60	M60		M60		M60	M60		M60		M60	M60			M60		M60	M60	
14	M60	M60		D40		M60		D40		M60	M60		D40		M60	M60		D40		M60	M60			M60		M60	M60	
15	M60	M60		M60		M60		M60		M60	M60		M60		M60	M60		M60		M60	M60			M60		M60	M60	
16	M60	M60		D40		M60		D40		M60	M60		D40		M60	M60		D40		M60	M60			M60		M60	M60	
17	M60	M60		M60		M60		M60		M60	M60		M60		M60	M60		M60		M60	M60			M60		M60	M60	
18	M60	M60	M60	D40		M60		D40		M60	M60	M60	D40		M60	M60	M60	D40		M60	M60	D40		D40		M60	M60	
19	M60	M60		D40		M60		D40		M60	M60		D40		M60	M60		D40				D40		D40		M60	M60	
20	D60	D40		D40		D40	D40	D40		D40	D40	D40	D40		D40	D40		D40		D40	D40	D40		D40		D40	D40	D40

●サンプル16

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
0																												
1																												
2																												
3																												
4																												
5																												
6																												
7																												
8																												
9																												
10																												
11																												
12									F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32									
13	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60
14	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60
15	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60
16	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60
17	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60
18	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60
19	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60
20	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60	M60

●サンプル25の画面設定

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
0	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32
1	F32																											F32
2	F32																											F32
3	F32																											F32
4	F32																											F32
5	F32																											F32
6	F32																											F32
7	F32																											F32
8	F32																											F32
9	F32																											F32
10	F32																											F32
11	F32																											F32
12	F32																											F32
13	F32																											F32
14	F32																											F32
15	F32																											F32
16	F32																											F32
17	F32																											F32
18	F32																											F32
19	F32																											F32
20	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32

●サンプル26

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
0	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32
1	F32																											F32
2	F32		H20																							H20		F32
3	F32		H20																							H20		F32
4	F32		H20	H10	H10	H10	H10	H10	H10	H10	H10	H10	H10	H10	H10	H10	H10	H10	H10	H10	H10	H10	H10	H10	H10	H20		F32
5	F32		H20	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	H20		F32
6	F32		H20	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	H20		F32
7	F32		H20	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	H20		F32
8	F32		H20	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	H20		F32
9	F32		H20	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	H20		F32
10	F32		H20	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	H20		F32
11	F32		H20	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	H20		F32
12	F32		H20	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	H20		F32
13	F32		H20	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	H20		F32
14	F32		H20	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	H20		F32
15	F32		H20	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	H20		F32
16	F32		H20	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	H20		F32
17	F32		H20	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	H20		F32
18	F32		H20	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	F70	H20		F32
19	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32
20	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32	F32

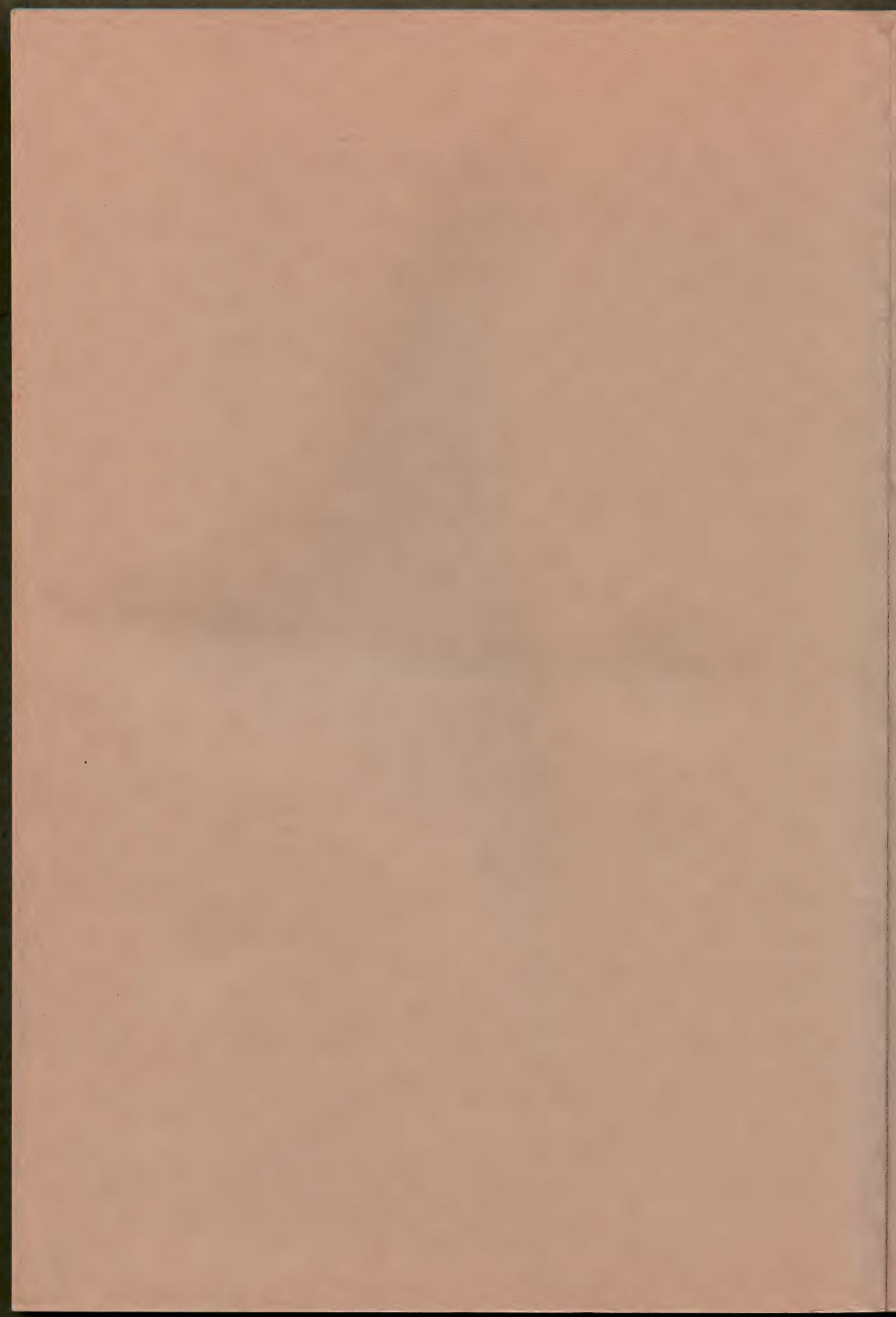
●サンプル27の画面設定

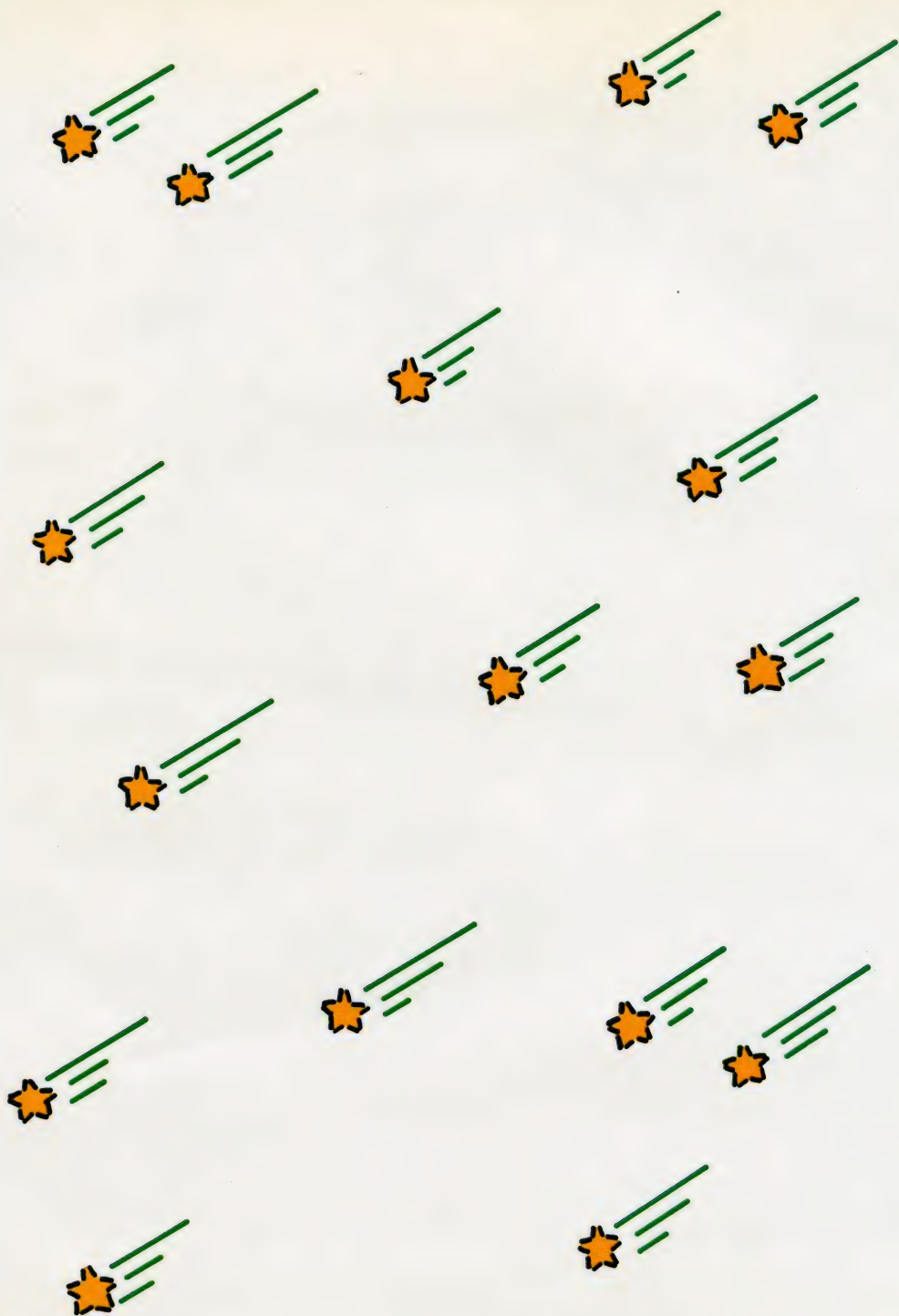
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27
0																												
1	M70	M70	M71	M71	M72	M72	M73	M73	M70	M70	M71	M71	M72	M72	M73	M73	M70	M70	M71	M71	M72	M72	M73	M73	M70	M70	M71	M71
2	M70	M70	M71	M71	M72	M72	M73	M73	M70	M70	M71	M71	M72	M72	M73	M73	M70	M70	M71	M71	M72	M72	M73	M73	M70	M70	M71	M71
3	M73	M73																									M72	M72
4	M73	M73																									M72	M72
5	M72	M72																									M73	M73
6	M72	M72																									M73	M73
7	M71	M71																									M70	M70
8	M71	M71																									M70	M70
9	M70	M70																									M71	M71
10	M70	M70																									M71	M71
11	M73	M73																									M72	M72
12	M73	M73																									M72	M72
13	M72	M72																									M73	M73
14	M72	M72																									M73	M73
15	M71	M71																									M70	M70
16	M71	M71																									M70	M70
17	M70	M70																									M71	M71
18	M70	M70																									M71	M71
19	M73	M73	M72	M72	M71	M71	M70	M70	M73	M73	M72	M72	M71	M71	M70	M70	M73	M73	M72	M72	M71	M71	M70	M70	M73	M73	M72	M72
20	M73	M73	M72	M72	M71	M71	M70	M70	M73	M73	M72	M72	M71	M71	M70	M70	M73	M73	M72	M72	M71	M71	M70	M70	M73	M73	M72	M72

一生懸命
ファミリーコンピュータ　オリジナルゲーム集25

製作＋著作——木村香奈枝＋斉藤千秋
企画協力——小牧自行
発行者——田村正隆
発行所——〔株式会社〕**ナツメ社**
東京都千代田区神田神保町1-52
電話〈03〉291-1257　振替＝東京3-58661
印刷——ラン印刷社
製本——三修紙工

写真撮影——丹野清志
製作協力——川村清＋まどか＋本多しのぶ＋内海純子＋高橋啓子
本文デザイン——中垣信夫＋早瀬芳文
イラスト——大崎吉之

ISBN4-8163-0561-0　●定価はカバーに表示してあります。

ファミリーコンピュータ

オリジナルゲーム集25

木村香奈枝
斉藤千秋 著

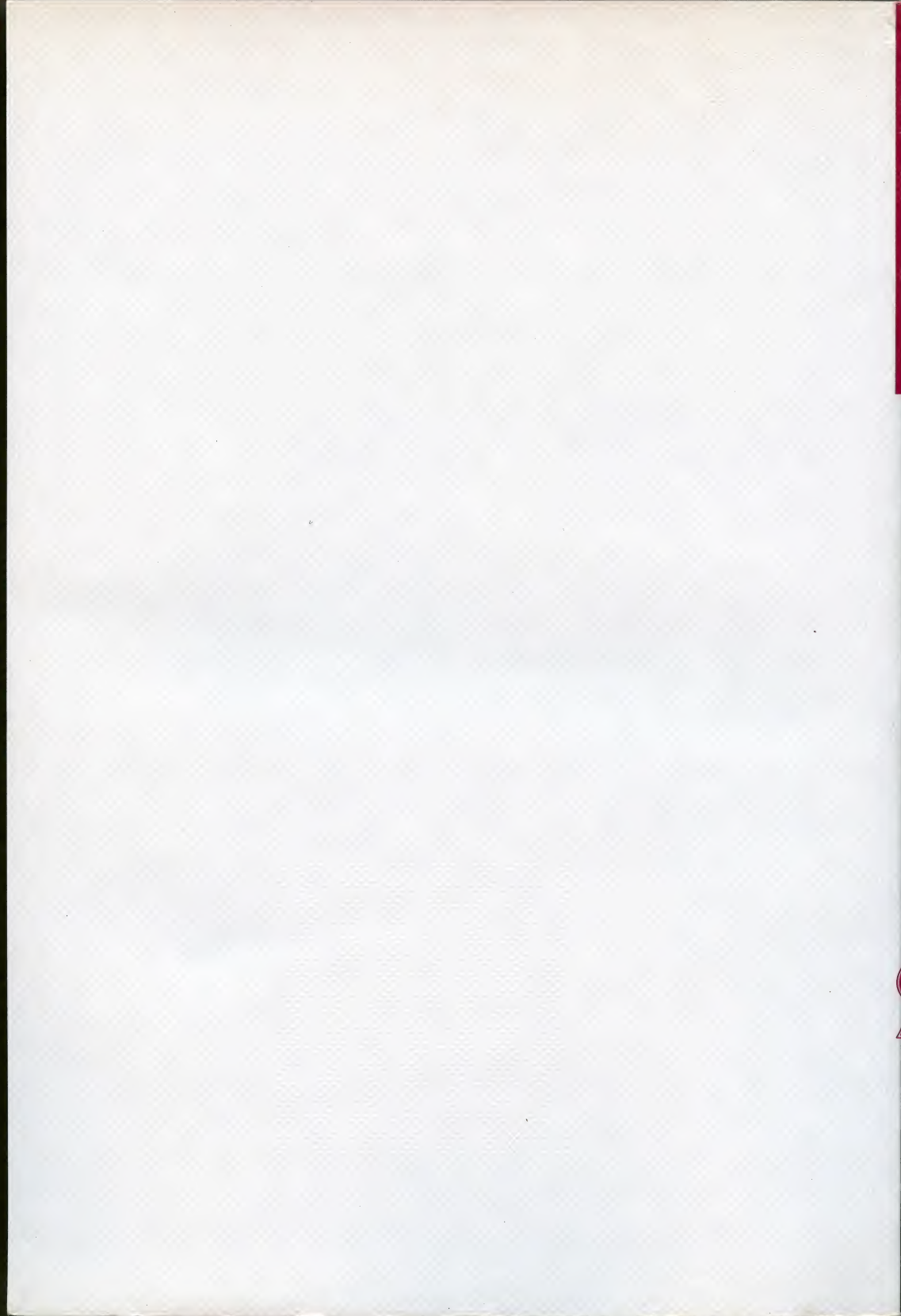
ナツメ社

定価980円　ISBN4-8163-0561-0　C2054　¥980E　ナツメ社

ファミリーコンピュータ™

オリジナルゲーム集

木村香奈枝・斉藤千秋著

ファミリーコンピュータ

オリジナルゲーム集25

木村香奈枝
斉藤千秋 著

ナツメ社

好評
発売中

ファミリーコンピュータ&
ベーシック パーフェクトブック ―980円

一生懸命ファミリーコンピュータ
ゲーム+ベーシック入門 ―980円

定価980円 ISBN4-8163-0561-0 C2054 ¥980E

25

木村香奈枝
斉藤 千秋 著

ナツメ社

25

木村香奈枝
斉藤　千秋　著

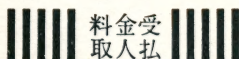
ナツメ社

ファミリーベーシック

ボクだけのゲームが25本もふえた!

うれし

ナツメ社

郵便はがき

料金受取人払

神田局承認

4830

差出有効期間
昭和61年8月
31日まで

101-

東京都千代田区
神田神保町一―五二

（受取人）

ナツメ社 行

ご氏名	（男・女） 歳
ご住所（〒　　　　）	
ご職業　会社員（事務・技術・管理職）　公務員（事務・技術・管理職） 自営業　教育　学生（大学・高校）　その他（　　　　）	
お買上げ書店名　　県 区 市　　町　　書店	

●ご購読ありがとうございました。
お手数ですが、このアンケートについてお答え下さい。

書　名

---

❶ 本書についてのご意見、ご感想をお聞かせください。(内容の程度、本の大きさ、価格など)。

❷ 今後、小社にどんなテーマの本をお望みですか。(○印を)

1）ディスク操作の解説書　2）ビジネスソフト集
3）ゲーム集　4）マシン語
5）言語（BASIC, COBOL, FORTRAN, PASCAL, C)
6）OS（MS-DOS, CP/M, UNIX, その他)
7）グラフィックス　8）データベース　9）ハード
10）パソコン通信　11）その他

❸ 質問❷で選択したテーマについて、その内容をもう少し具体的にお聞かせください。

❹ 現在パソコンをお持ちですか？

■ハイ
1）機種名
2）どんなことにお使いですか（できたら具体的に)。
3）あなたのパソコン歴は
■イイエ